RUBRIQUE POQUETTES

TRACE

TIREZ PLUS DE VOTRE TRS-80*

NUMERO 2

Sommaire, p 3 • Coup d'œil sur quatre ordinateurs, p 17 • Essais : EDTASM + ; Adventure, p 20 • Copiez nos programmes sans erreur, p



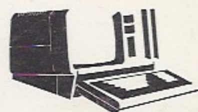
27 • Décidez avec Visicalc, p 31 • Chez



un transporteur, p 33 • Un programme d'enseignement, p 35 • Le décathlon, p 38 • La boîte à outils mathématiques, p 39 • Jeu : Mathic, p 42 • Modèles 1 et



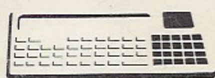
3, quelles différences? p 44 • Prévoir, p 47 • Nevada Cobol, p 50 • CP/M sur modèle 1, p 52 • La représentation des variables, p 53 • Pour vous



aider à programmer, p 55 • 20 francs pour connecter deux TRS, p 57 • Sonorisez votre TRS, p 58 • La multiplication égyptienne, p



59



• RUBRIQUE POQUETTES: tracé de courbes, p 60; un mini-traitement de textes, p 61; les cavernes, p 62; la "calculette", p 64

N° 2 - 2e trimestre 1982

Canada : 6.50 $ - Belgique : 200 FB - Suisse : 11 FS 25F

DÉSORMAIS VOS MICRO-ORDINATEURS SONT DE VÉRITABLES SYSTÈMES DE GESTION AVEC CORVUS

— TECHNOLOGIE **WINCHESTER**
— Capacité de stockage : 5, 10, 20, 40 et 80 méga-octets,
— Compatible avec APPLE II et III, LSI 11 BUS S-100, **HP 125,** COMMODORE, **XEROX 820,** ZENITH H/89, TRS-80 II et III, **NEC, IBM ordinateur personnel,** ATARI...
— CONSTELLATION : multiplexeur intelligent pour partager la mémoire de masse et une imprimante entre plusieurs micro-ordinateurs.
— MIROIR : sauvegarde rapide et économique sur magnétoscope jusqu'à 80 méga-octets.
— OMNINET : réseau local sur une simple paire torsadée de **1 200 mètres** (connexion jusqu'à 64 micro, à l'aide de Transporteurs de faible coût. (Prix : 6 250 F (ht).

PRIX H.T. : 32 835 F







CORVUS SYSTEMS
MICROLOGIE
importateur exclusif pour la France
143 ter, avenue J.-B.-Clément, 92100 BOULOGNE - Tél. 604 78 56

TRACE

n°2

Sommaire

Editorial	3
Courrier	11
Magazine	15
Coups d'oeil	
sur le LNW-80	17
sur le Modèle 16	19
sur les poquettes	19
Essai: EDTASM+	20
Essai: Adventure	21
Librairie	22
Trucs à Brac	24
Côté Court	26
Somme de Contrôle	27
VisiCalc dans le bâtiment	31
Les TRS roulent pour vous	33
Programme pédagogique	35
Essai:Olympic Decathlon	38
Essai: Mumath	39
Mathic	42
Différences entre les modèles 1 et 3	44
Il vous faut prévoir	47
Essai: Nevada Cobol	50
Essai: CP/M avec le Mapper d'Omikron	52
Les variables et leur représentation	53
Un programme moniteur	55
La boîte de cachous	57
Un ampli pour votre modèle 1	58
La multiplication égyptienne	59
Tracez des courbes	60
Un mini-traitement de textes pour poquette	61
Aventure sur poquette	62
Vérifiez vos additions	64
Le DOS de A à Z	65
Index des mots-clés	66

Editorial

Nous pensions avoir pris toutes nos précautions pour pouvoir répondre efficacement aux demandes de nos lecteurs, tant de renseignements que d'abonnements.

Mais nous avions sérieusement sous-estimé l'accueil que vous feriez à notre premier numéro! Alors, nous avons été débordés, mais les choses sont maintenant (à peu près) rentrées dans l'ordre. Nous avions un instant espéré gérer les abonnements sur ordinateur: impossible en fait de mettre le fichier des abonnés sur une seule disquette, il en aurait fallu 4 ou 5 de plus, et cela n'aurait de toute façon pas résolu le problème! Les abonnements sont donc gérés à la main, ce qui devrait au moins pour quelque temps limiter les erreurs.

Il nous a bien sur été impossible de répondre individuellement à toutes vos lettres, mais sachez qu'elles nous ont fait grand plaisir. Vous, nos lecteurs, vous êtes tolérants, pas bougons du tout devant nos erreurs, et la confiance que vous avez été nombreux à nous accorder en vous abonnant sans même avoir vu le premier numéro en est le meilleur exemple. Mais ne croyez pas que nous en profiterons pour continuer à sortir en retard!

Ce numéro déjà doit beaucoup à votre courrier: certains articles sont ceux que vous nous avez envoyés, et nous avons essayé de tenir compte de vos suggestions.

Ainsi, vous trouverez dans ce numéro 2 une rubrique pour les "poquettes" signalée par la bordure bleue des pages en fin de journal. Si vous n'avez pas de poquette ne négligez pas pour autant de lire cette rubrique: les programmes qui y figurent ne déshonorent pas les ordinateurs individuels "de table".

Les débutants se disaient lésés par manque de programmes pour leurs machines 16K: nous leur proposons cette fois de petits programmes, voire des PUL (programmes d'une ligne!) qui devraient satisfaire même les possesseurs d'une mémoire plus confortable.

Et puis, nous avons le plaisir de vous présenter une première mondiale (si, si!): le moyen de vérifier si vous avez recopié correctement les programmes que nous publions, grâce à un système de somme de contrôle. Maintenant, si un programme que vous avez tapé ne marche pas, vous pourrez être sûr que cela ne vient pas de vous, mais (hélas!) de nous! Ce qui nous incitera bien sûr à faire encore plus attention, soyez-en assurés.

TRACE

* TRS-80 est une marque déposée de Tandy Radio-shack.

Rédacteur en chef : Alain Pinaud - *Editeur :* Bernard Savonet - *Directeur de la publication :* Jean-Pierre Nizard - *Rédaction et abonnements :* Editrace, 8 rue Saint-Marc, 75002 Paris. - *Régie publicitaire :* Force 7, 41 rue de la Grange-aux-Belles, 75483 Paris Cedex 10 - Tél. (1) 238 66 10. - *Diffusion auprès des boutiques informatiques et libraires :* Editions du PSI, 41 rue Jacquard - BP 86 - 77400 Lagny - Tél. (6) 007 59 31.

Abonnements 4 numéros (un an) : France 85 FF; Belgique 680 FB; Suisse 37 FS; Canada 22 $C; Etranger 100 FF.

Ont collaboré à ce numéro : Kaarina Alain, Tristan d'Amico, Jean-Luc Bérardo, Jean-David Blanc, Jean-Claude Bouman, Michel Claveau, Sylvine Dautref, Charles Feydy, Henri Frère, Wolfgang Lauter, Jean Le Flour, Alain Marmoiton, Steen Pedersen, Jean Lech Saconney, Roger Sawicki, Richard Schomberg, Stéphane, J.-P. Stremler, Jean-Paul du Tillet, René Thomas.

un bijou d'imprimante à 2490F t.t.c



Photographie non contractuelle.

Serdenir & Associés

Avec l'arrivée sur le marché de la GP-100, l'impression par impact (méthode UNI-HAMMER de SEIKO) va encore faire des ravages. Ce procédé, d'un coût de fabrication imbattable, permet enfin à tout amateur de micro-informatique de posséder sa propre imprimante à un prix plus que raisonnable. Connectable à presque tous les micro-ordinateurs (idéale pour TRS-80™, Apple II™, Pet™), elle est munie en standard d'un interface Centronics. Elle imprime sur du papier ordinaire de 254 mm. (h) par 241 mm. (1) jusqu'à 1 original plus 2 copies à la vitesse de 30 caractères par seconde. Elle vous permet le choix entre deux largeurs de caractères (80 caractères par ligne en normal et 40 en élongué pour les titres) et un mode ***graphique point par point*** *où vous laisserez libre choix à votre imagination jusqu'à concurrence de* ***480 points par ligne.*** *Elle met à votre disposition la totalité des 116 caractères standard US-ASCII en matrice 5 x 7, plus les caractères allemands, suédois et* ***français accentués.*** *Livrée avec un manuel français et une cassette de ruban encreur, elle est* ***garantie 90 jours*** *pièces et main-d'œuvre. D'un poids faible (moins de 4,5 kg), compacte (234 mm en hauteur, 420 mm en largeur et 136 mm en hauteur), de consommation réduite (5W au repos, 25 W en impression), elle est le complément* ***indispensable à votre système,*** *et nous serons heureux de vous la faire connaître à l'un de nos magasins,* ***où notre personnel de vente vous accueillera avec le sourire.***

TRS-80, Apple II™ et Pet™ sont les marques déposées respectives de TANDY CORPORATION, APPLE COMPUTER et COMMODORE BUSINESS MACHINES.

Nous vous réservons le meilleur accueil à nos boutiques ouvertes du lundi au samedi de 9h30 à 12h30 et de 14h à 19h.

Vente par correspondance - Crédit - Carte bleue - Détaxe à l'exportation - Expédition dans toute l'Europe.

281.23.17 + 92, rue St Lazare, galerie 92, 75009. Paris
858.15.95 + Centre Commercial Terminal 93, 93100. Montreuil

GRAPHIE FRANCE



TRACE

le magazine des utilisateurs de TRS-80*



TRS-80 modèle 1, TRS-80 modèle 2, TRS-80 modèle 3, TRS-80 Couleur, TRS-80 de poche, Sharp* PC-1211 Vidéo-Génie*, LNW* : si vous utilisez ou si vous comptez acheter un de ces ordinateurs, sachez que la revue TRACE a été créée pour vous. TRACE vous fournit quatre fois par an des programmes, des astuces, de nouvelles idées d'applications. TRACE est indépendant de tout constructeur, et vous tient au courant de toutes les nouveautés dans les domaines des logiciels, des matériels et des périphériques, quelle qu'en soit la source. TRACE teste pour vous, en toute objectivité et indépendance, les produits matériels ou logiciels qui vous intéressent. TRACE n'est pas en vente chez les marchands de journaux. Pour vous abonner ou pour recevoir un numéro, il vous suffit de nous retourner le bon de commande ci-dessous.

*TRS 80, Sharp, Vidéo-Génie et LNW sont des marques déposées.

TRACE, le complément indispensable de votre TRS-80

BON DE COMMANDE

à retourner à TRACE, 8 rue Saint-Marc 75002 PARIS

TR 2

Nom ______________________ Profession ______________

Adresse ____

Pays ______________ Code postal ______________ Ville ______________

☐ Je désire recevoir le n° 1 ☐ le n° 3 ☐ de TRACE
(Prix d'un n° 25 FF; Belgique 200 FB; Suisse 11 FS; Canada 6,50 $C; Etranger 30 FF).

☐ Je désire m'abonner à TRACE pour 1 an, 4 nos, à partir du n° 3
(Tarif France 85 FF; Belgique 680 FB; Suisse 37 FS; Canada 22 $C; Etranger 100 FF).

Ci-joint mon règlement indispensable par chèque bancaire ☐ chèque postal ☐

(Pour les pays autres que la France, utiliser un virement en FF compte Crédit Lyonnais Paris n° 30002 00402 8401 M. Les frais de virement sont à la charge de l'acheteur.)

Video Genie System

GENIE I EG 3003

SON ~ MINUSCULES ~ BASIC (R)*

*** Extension BASIC microsoft LEVEL II compatible TANDY.**



EG 3003

- 16 K RAM Utilisateur
- 14 K ROM BASIC Microsoft LEVEL II
- Microprocesseur Z 80
- Modulateur vidéo (Sortie UHF 625 lignes)
- Clavier QWERTY
- Magnétophone à cassette intégré au boîtier, pas de réglage de volume
- Prise DIN pour deuxième magnétophone
- Ecran 16 lignes 32 ou 64 caractères
- Graphismes 128 x 48
- Cassettes et programmes compatibles avec TRS 80* Level II
- Alimentation intégrée 110 / 220 / 240 V 50 Hz
- Branchement direct sur téléviseur ou moniteur vidéo
- Livré avec: cordons, 1 cassette démonstration. Moniteur en option
- Bus compatible TRS 80*
- Vu-mètre, réglage niveau de lecture

EG 3008

- Mêmes caractéristiques + clavier numérique minuscules + software RS 232 C + clavier fonction. Sortie vidéo seulement

*** TRS 80 marque déposée «Tandy Radio Shack».**

NOUVEAU: BASIC avec renumérotation — Instruction Hard Copy - Moniteur en langage machine - Nouvelle routine clavier avec minuscules - Répétition curseur clignotant - Sortie son avec haut-parleur intégré.

OPTIONS

- Imprimante graphique incrémentale TONO HC 900, tracteur et friction 40 - 48 - 80 - 96 - 136 colonnes
- Imprimante TONO HC 800, 80 - 132 colonnes 120 CPS
- Boîte d'expansion EG 3014
- Moniteur professionnel écran vert TONO CRT 120 G

EG 3014 - 16 (32)

Boîte d'expansion comprennant : contrôleur de disques Floppy, interface parallèle Centronics, mémoire RAM 16 K (32 K). En option, interface RS 232 C, Bus S 100. Fourni avec câble de raccordement à EG 3003 ou EG 3008

EG 3016

Interface de raccordement pour imprimante type Centronics (TONO HC 800) avec câbles.

EG 400 T

Simple lecteur de disque.

EG 401 AT

Double lecteur de disque.

EG 3021

Doubleur de densité 211 K par disque

EG 3020

Adaptation Bus RS 232 C

EG 3022

Adaptation Bus S 100

EG 3018

Floppy câble

EG 3016 C

Câble imprimante

~~4.350 F~~

4.150 F TTC



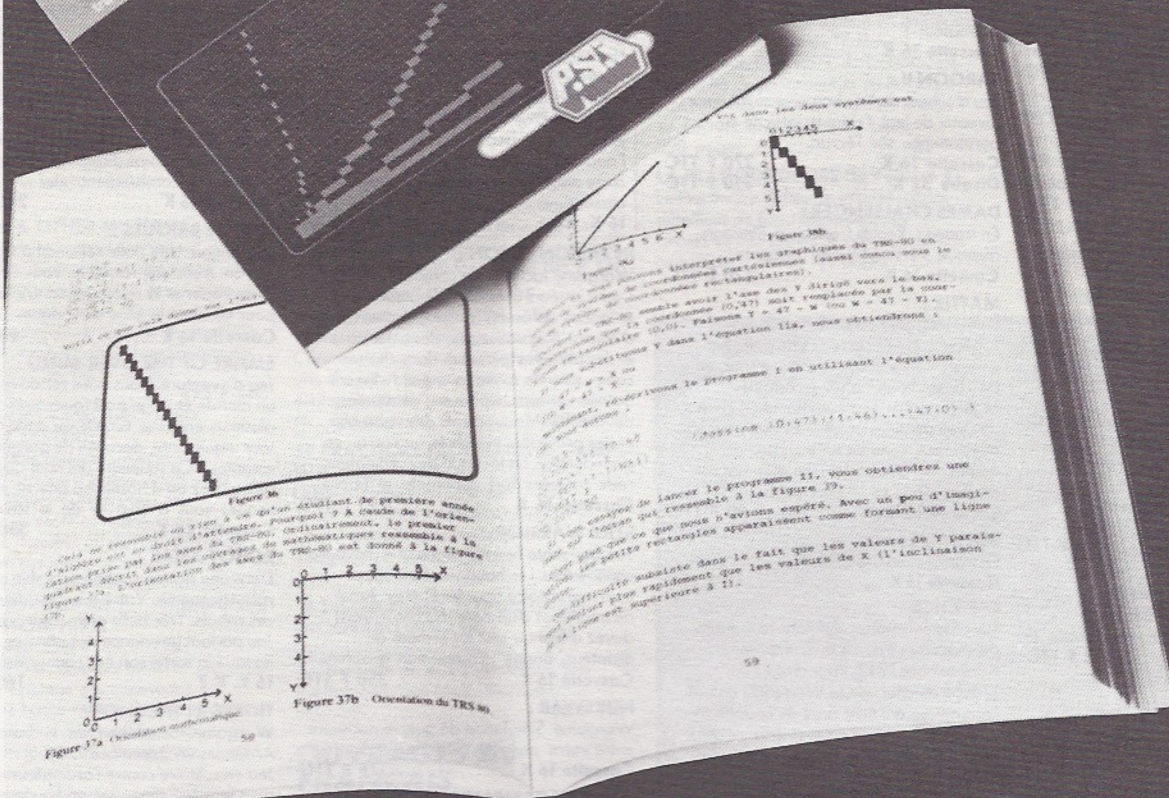
GENERALE ELECTRONIQUE SERVICES

68 ET 76, AVENUE LEDRU ROLLIN, 75012 PARIS
TEL. : 345.25.92 - TELEX : 215 546 GEFPAR

INDISPENSABLE :

LES GRAPHIQUES SUR TRS 80

TRADUIT PAR P.S.I.



Les graphiques sur TRS-80
par Don Inman
traduit par Alain Pinaud



Dans ce livre, l'auteur montre comment, avec une connaissance minimale du Basic et du TRS-80, visualiser des graphiques sur écran. Il débute par les notions élémentaires de tracés de ligne puis aborde les dessins géométriques pour déboucher sur l'animation de dessin et autres applications plus complexes.

144 pages - 75,00 FF /570,00 FB

A.GRAPH



Envoyer ce bon accompagné de votre règlement à **EDITIONS DU P.S.I.** ou à **P.S.I. BENELUX** I.TR.1

Je désire recevoir ________ exemplaires du volume LES GRAPHIQUES SUR TRS 80 au prix unitaire de 75,00 FF (570,00 FB) taxes, port et emballage compris (par avion : ajouter 5 FF (44 FB) par livre).

TOTAL ________

NOM ____________________ PRENOM ____________________

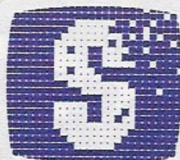
rue ____________________ N° ______

Code post. |_|_|_|_|_| Ville ____________________

Editions du P.S.I.
41-51, rue Jacquard
BP 86 - 77400 Lagny-s/Marne
Téléphone (6) 007.59.31

P.S.I. BENELUX
5, avenue de la Ferme Rose
1180 Bruxelles
Téléphone (2) 345.08.50

VIDEO-GENIE TRS



SIVEA

31, Bd DES BATIGNOLLES 75008 PARIS
TEL. : 522.70.66 (+) TELEX 280 902 F

DETAXE A L'EXPORTATION

Ouvert sans interruption du lundi au samedi de 9 h 30 à 18 h 30
Métro : Rome, Place de Clichy, Europe - Parking assuré au 43 bis, Bd des Batignolles
Vente par correspondance - Crédit - Leasing - Carte Bleue - Visa.

JEUX TRS 80* ET VIDEO-GENIE

ARRIVAGE DE NOUVEAUTES TOUTES LES QUINZAINES

TOUS CES LOGICIELS FONCTIONNENT SUR TRS 80 MODELE 1 ET MODELE 3 AINSI QUE SUR VIDÉO-GÉNIE 1 ET 2.

CONFLICT 2500 :
Jeu de type wargame. Simulation d'un combat galactique au 25e siècle. Plusieurs scénarios. Jeu pour plusieurs joueurs ou en solitaire contre l'ordinateur.
16 K, Cassette .... 175 F TTC

LES ENVAHISSEURS :
En français. Toute une flotte de vaisseaux ennemis s'approche de votre base. Détruisez les avant qu'ils ne vous atteignent ! Belle animation. Sonore.
Cassette 16 K .... 110 F TTC

ATTACK FORCE :
Combat de vaisseaux galactiques dans un labyrinthe d'étoiles. Belle animation graphique, sonore. Fonctionne avec commandes au clavier ou avec un paddle.
Cassette 16 K .... 160 F TTC

COSMIC FIGHTER :
Seul à bord de votre vaisseau cosmique, vous devez détruire la flotte des envahisseurs extra-galactiques. Des groupes d'attaques sont envoyés sur vous et vous pilonnent sans merci. Belle animation graphique, sonore. Fonctionne avec commandes au clavier ou avec un paddle.
Cassette 16 K .... 160 F TTC

COMBAT GALACTIQUE :
En français. Même chose que "Les envahisseurs" mais en plus, certains vaisseaux ennemis se regroupent et foncent sur nous en tirant.
Cassette 16 K .... 130 F TTC

ETOILE NOIRE :
En français. Des vaisseaux ennemis traversent l'écran à toute vitesse. Détruisez-les en vol tout en évitant leur tir.
Cassette 16 K .... 90 F TTC

SUPER NOVA :
Vous traversez une partie de la galaxie encombrée d'astéroïdes. Saurez-vous les éviter ou les détruire à temps ? Belle animation graphique. Fonctionne aussi avec Paddles.
Cassette 16 K .... 160 F TTC

METEOR MISSION 2 :
Des astronautes sont en détresse sur une planète entourée d'une ceinture d'astéroïdes. Avec une navette spatiale, descendez les chercher un par un en évitant ou en détruisant les astéroïdes. Jeu sonore avec d'excellents graphismes. Possibilité de jeu avec paddles !
Cassette 16 K .... 160 F TTC

IAGO (OTHELLO CHALLENGER) :
Jeu d'Othello. Excellent niveau. Très belle présentation graphique. A ne pas essayer si vous craignez d'être humilié par votre ordinateur !
Cassette 16 K .... 245 F TTC

SARGON II :
Jeu d'échecs de très bon niveau (plusieurs niveaux de jeu). Echiquier affiché en permanence sur l'écran.
Cassette 16 K .... 270 F TTC
Disque 32 K .... 310 F TTC

DAMES CHALLENGER :
En français. Règles françaises. Plusieurs niveaux. Version très performante.
Cassette 16 K .... 220 F TTC

MATTIX :
En français. Jeu de réflexion contre l'ordinateur. Sur l'écran : une grille de chiffre (− 10 à + 10). A chaque tour, vous devez en choisir un selon certaines règles, puis l'ordinateur joue et en choisit un aussi selon les possibilités que vous lui avez laissé. Ainsi de suite jusqu'à épuisement des chiffres. Le gagnant est celui qui totalise le plus de points.
Cassette 16 K .... 90 F TTC

SOLITAIRE :
En français. Jeu du solitaire.
Cassette 16 K .... 60 F TTC

TANKTICS :
Superbe wargame simulant la bataille de chars aux abords du Dniepr le 6 Septembre 1942. Vous dirigez un groupe de chars de la Wermacht contre l'ordinateur qui tient ceux de l'armée Rouge soviétique. Règles classiques du wargame sur cases hexagonales. (Effets de terrain, etc.). Utilise toutes les possibilités du wargame sur ordinateur : l'effet d'un coup au but est différent selon que le char touché se présente de face (blindage le plus épais) ou de flanc (blindage plus mince). Possibilité de voir sans être vu, etc... Très belle présentation en coffret carton comprenant un plan de jeu, des pièces cartonnées pour figurer les unités, un manuel d'instructions et une cassette. 5 scénarios différents. Un jeu Avalon Hill.
16 K, K 7 .... 295 F TTC

THE BATTLE OF SHILLOH
Très bon wargame simulant la bataille de Shilloh le 6 Avril 1862, durant la guerre civile américaine. Règles classiques du wargame (zones de contrôle, effets de terrain, etc). Jeu contre l'ordinateur qui vous laisse choisir votre camp en début de partie. Plan de jeu affiché en permanence à l'écran. Fournit en coffret carton contenant une carte plastifiée, un manuel et une cassette.
16 K, K7 .... 275 F TTC

INVASION ORION :
Wargame tactique de combat spatial. 1 ou 2 joueurs contre l'ordinateur. Dix scénarios différents (possibilité d'en créer d'autres). Des vaisseaux, des astéroïdes et des planètes (plusieurs dans chaque camp, selon le scénario choisi) se livrent bataille à coup de phasers, de missiles, de torpilles. Mouvements des vaisseaux et des projectiles visualisés sur l'écran en permanence. La tactique de l'ordinateur reste toujours fort cohérente et habile.
Cassette 16 K .... 295 F TTC

STAR WARRIOR :
Sur la planète Fornax règne une dictature impitoyable. Les habitants font appel à un mercenaire pour les libérer. Armé de missiles et d'un désintégrateur, vous devez affronter seul les armées du dictateur. Bonne présentation graphique.
Cassette 16 K .... 350 F TTC

NUKEWAR :
Wargame. Simulation de guerre nucléaire entre votre pays et celui de l'ordinateur.
Cassette 16 K .... 175 F TTC

MIDWAY CAMPAIGN :
Wargame. Simulation de la bataille de Midway. L'ordinateur joue pour le camp japonais.
Cassette 16 K .... 175 F TTC

LUNAR LANDER :
Très belle animation graphique d'alunissage. Demande beaucoup d'adresse.
Cassette 16 K .... 160 F TTC

FLIGHT SIMULATOR :
Simulation de vol en temps réel. Sur l'écran : tous les cadrans du tableau de bord et une vue en 3 dim. du paysage. Apprenez à piloter votre avion, puis partez attaquer les bases ennemies.
Cassette 16 K .... 270 F TTC

AIR TRAFFIC CONTROLLER :
Très bonne simulation du contrôle de l'espace aérien de deux aérodromes. Jeu en temps réel. 27 avions et jets se fient aveuglement à vos directives. Avions en détresse, encombrement, etc.
Cassette 16 K .... 245 F TTC

MUR DE BRIQUES :
En français. Une balle rebondit à travers l'écran. Avec une raquette, vous devez la rattraper et la renvoyer contre le mur pour éliminer le plus de briques possible.
Cassette 16 K .... 90 F TTC

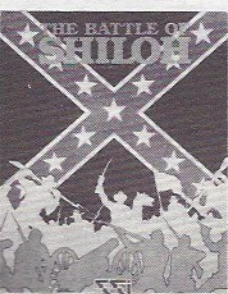
EMPIRE OF THE OVER-MIND :
Jeu d'aventure. Vous vous retrouvez dans un monde étrange où il faudra résoudre plusieurs énigmes. Quelle est donc cette tour inquiétante, décorée de gargouilles vivantes ? Ce squelette dansant que vous rencontrez est-il hostile ou amical, et que risquez-vous à essayer de le savoir ?
Cassette 48 K .... 350 F TTC

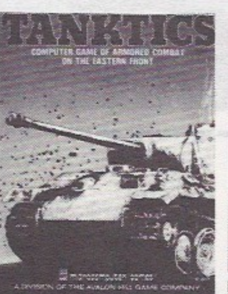
ROBOT ATTAK :
Dans une station spatiale envahie par des robots ennemis. Votre mission : détruire ces robots. Très belle animation graphique. Jeu parlant (prononce des phrases lorsque la sortie son est connectée).
16 K, K 7 .... 160 F TTC

TIGERS IN THE SNOW
Wargame. Simulation de la bataille des Ardennes en Décembre 1944. Jeu en solitaire contre l'ordinateur qui peut tenir le camp allemand ou le camp allié. Carte affichée en permanence à l'écran. Très bon wargame .... **275 F TTC**

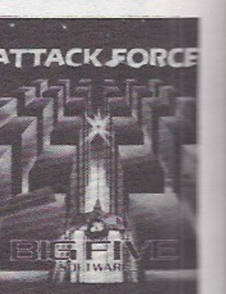












*TRS 80 est une marque déposée par TANDY-RADIO-SHACK NOS PRIX SONT DONNES A TITRE INDICATIF ET PEUVENT ETRE MODIFIES SANS PREAVIS

VIDEO-GENIE TRS

DETAXE A L'EXPORTATION

SIVEA

31, Bd DES BATIGNOLLES 75008 PARIS
TEL. : 522.70.66 (+) TELEX 280 902 F

Ouvert sans interruption du lundi au samedi de 9 h 30 à 18 h 30
Métro : Rome, Place de Clichy, Europe - Parking assuré au 43 bis, Bd des Batignolles
Vente par correspondance - Crédit - Leasing - Carte Bleue - Visa.

LIBRAIRIE TRS 80*ET VIDEO-GENIE

ARRIVAGE DE NOUVEAUTES TOUTES LES QUINZAINES

OUVRAGES EN ANGLAIS :

THE BOOK Volume 1 :
Toutes les routines mathématiques de la ROM du TRS (ou Vidéo-Génie) et comment s'en servir dans un programme en langage machine : addition, soustraction, multiplication et division. SIN, COS, TAN, LOG, INT, etc... Localisation des RAM de travail de la ROM. 120 pages .... **135 F TTC**

THE BOOK Volume 2 :
Mode d'emploi des routines d'entrée sortie de la ROM du TRS (ou du Vidéo Génie) pour utilisation dans des programmes en langage machine. Gestion du clavier de l'écran du lecteur de cassettes, etc. De nombreux "trucs" : par ex. : comment désactiver la touche "BREAK". 120 pages .... **135 F TTC**

TRS DISK & OTHER MISTERIES :
Tout sur le fonctionnement "intime" du D.O.S. : localisation et interprétation du DIRECTORY, récupération de données et de programmes sur une disquette endommagée, dévérouillage des mots de passe, etc. Explication du fonctionnement des utilitaires du NEW DOS. 130 pages .... **195 F TTC**

MICROSOFT BASIC DECODED & OTHER MISTERIES :
Ouvrage très complet sur la ROM Microsoft : Mode d'emploi et adresses des principales routines utiles en assembleur : routines mathématiques, entrée de données au clavier, affichage sur l'écran et sur imprimante. Liste complète des points d'entrée et de sortie de la ROM. 300 pages .... **260 F TTC**

THE CUSTOM TRS & OTHER MISTERIES :
Vous apprend comment modifier votre TRS pour : obtenir des minuscules, un graphisme haute-résolution, une horloge temps réel, etc... Avec schémas et instructions (partie non compatible Vidéo-Génie).
Dans la seconde partie, après le "hard" : le "soft" (compatible Vidéo-Génie) : comment écrire des programmes Basic qui s'auto-exécutent, comment initialiser le MEMSIZE, comment compresser les lignes d'un programme, etc. 290 pages . **PRIX NC**

BASIC FASTER AND BETTER & OTHER MISTERIES :
Comment programmer de façon structurée en Basic. Comment utiliser ensemble le Basic et le langage machine. Manipulations de bits sous Basic. De très nombreux "trucs" concernant notamment l'utilisation du clavier et de la Vidéo. Nombreuses routines (tri, recherche, etc.) en langage machine, à utiliser sous Basic. Tout un chapitre est en outre réservé au TRS modèle 2. 288 pages .... **280 F TTC**

TRS 80 INTERFACING Volume 1 :
Le TRS 80 et la communication avec son environnement. Nombreuses expériences décrites. Schémas. (Non compatible Vidéo-Génie) .... **105 F TTC**

TRS 80 INTERFACING Volume 2 :
Complète le Volume 1 : Conversions analogique - digital, traitement des données. Communication série. Gestion des interruptions. Expériences et schémas. (Non compatible Vidéo-Génie) **105 F TTC**

TRS ASSEMBLY LANGUAGE :
La programmation du TRS (ou du Vidéo-Génie) en langage machine. Comment programmer une multiplication, une division, faire une lecture ou une écriture cassette, un accès disque, etc... 186 pages .... **140 F TTC**

57 PRACTICAL PROGRAMS & GAMES IN BASIC :
Listings complets en Basic de 57 programmes mathématiques et de jeu : intégration numérique, interpolation linéaire, space wars, etc... .... **90 F TTC**

24 TESTED READY TO RUN GAME PROGRAMS IN BASIC :
Listings complets en Basic de programmes de jeux : Star Warp, Sink the Bismarck, Sub Hunt, Biorhytm, etc. .... **90 F TTC**

OUVRAGES EN FRANÇAIS

MANUEL D.O.S. ET NEW - D.O.S. en français :
Utilisation des drives, programmation en Disk Basic, emploi du Disk Operating System et des utilitaires : COPY, SUPERZAP, emploi de l'Editeur. Assembleur disques (EDTASM), du désassembleur (DISASSEM), etc. 92 pages .... **95 F TTC**

LA PRATIQUE DU TRS 80 Volume 1 :
Programmation et utilisation du TRS 80 en Basic. Jeu d'instruction du Basic. Exemples 128 pages .... **65 F TTC**

LA PRATIQUE DU TRS 80 Volume 2 :
Fonctions spéciales du Basic (USR, PEEK, POKE, etc.). Programmation en langague machine. Utilisation de l'Editeur. Assembleur, du T-BUG, du DEBUG, etc. Exemples de programmes en langage machine, notamment : tracé de droites sur l'écran, instructions du Z 80. 220 pages .... **85 F TTC**

LA PRATIQUE DU TRS 80 Volume 3 :
Le Hardware. Schémas et explications détaillées de fonctionnement des circuits du TRS (ouvrage non compatible avec Vidéo-Génie). 128 pages . . **75 F TTC**

LE BASIC ET SES FICHIERS Volume 1 :
Création, utilisation de fichiers sous langage Basic. Nombreux conseils et "trucs" pour optimiser la gestion des fichiers sur micro-ordinateur. 144 pages .... **75 F TTC**

LE BASIC ET SES FICHIERS Volume 2 :
Exemples de programmes utilitaires pour la gestion des fichiers en Basic : générateur de saisie d'écran, tri rapide, etc... 144 pages .... **75 F TTC**

PROGRAMMER EN BASIC :
Très bonne initiation à la programmation des micro-ordinateurs. 132 pages .... **65 F TTC**

PROGRAMMER EN ASSEMBLEUR :
Introduction complète à la programmation en langage machine (z 80). De très nombreux exemples et exercices résolus. 144 pages .... **75 F TTC**

LA PRATIQUE DU TRS 80 BASIC 2 :
Manuel très complet de programmation du TRS 80. Explications claires et détaillées. Nombreux exemples : **70 F TTC**

MICROSOFT BASIC DECODED & OTHER MYSTERIES for the TRS-80

BASIC FASTER AND BETTER & OTHER MYSTERIES

La pratique du TRS-80

*TRS 80 est une marque déposée par TANDY-RADIO-SHACK

TRA 02

BON DE COMMANDE

Retourner à : SIVEA S.A. - 31, Bd des Batignolles 75008 PARIS

Nom : ______ Prénom : ______
Adresse : ______
Ville : ______
Code Postal : ______ Bureau Distributeur ______
Pays : ______ Téléphone : ______

Quantité	DESIGNATION	PRIX UNIT.	PRIX TOTAL

MODE DE REGLEMENT :	TOTAL
Chèque bancaire joint ☐	Participation frais de port et d'emballage + 30 F. Voir ci-dessous.
CCP joint ☐	Contre-Remboursement : + 30 F (France seulement)
Mandat-lettre joint ☐	Etranger et DOM-TOM : + 30 F
Contre-Remboursement ☐	TOTAL

FRAIS DE PORT ET D'EMBALLAGE.
Ajouter **220 F** pour toute commande comportant, pour tout ou partie, des articles tels que : unité centrale, floppy-disk, imprimante, moniteur vidéo, papier pour imprimante. Transport par service express avec assurance comprise. Corse, DOM-TOM, étranger : nous consulter par téléphone ou courrier en ce cas).

courrier du lecteur

HOURRA !

- Enfin, voilà notre revue TRS ! Bravo ! (Irénée Prat).

- C'est avec un grand plaisir que j'apprends la création d'une revue destinée aux TRS-80 ! Je vous transmets mes plus vifs encouragements et mes voeux de longue route à TRACE ! (Christian Stéfani).

- Je vous félicite vivement pour le premier numéro de TRACE (JP. Stremler).

- Votre publication m'intéresse vivement ... et était très attendue chez les amateurs francophones (C. Corbou).

TRACE en retard ...

- Je vous ai commandé, il y a environ 3 semaines, le numéro 1 de "TRACE" et aujourd'hui, soit environ 1 mois après, je n'ai toujours rien reçu. S'agit-il d'un oubli, d'une négligence, d'un retard ? Comprenez ma déception d'autant plus grande que mon idée première était que des personnes voulant faire partager les plaisirs du TRS ne pouvaient qu'être sérieuses ... (Pierre Ardaud).

* Vous avez raison. Le facteur lui aussi, lorsqu'il nous apporte chaque jour deux gros sacs de colis postaux, nous prend de moins en moins au sérieux ! Mais il est moins évident de dépouiller 2000 lettres que d'en envoyer une. En réalité, nous avons été "légèrement" dépassés par les évènements, et la revue est sortie avec un mois de retard par rapport à l'annonce qui en était faite. Pour nous faire pardonner, il nous a fallu très tôt penser SERIEUSEMENT au numéro 2 ...

... d'un an !

- Je vous joins la photocopie du dernier bulletin de l'AUT où il est fait mention qu'une revue nommée TRACE, consacrée au TRS-80, serait publiée depuis Janvier 81 (JC. Ruby).

* Un mois de retard, c'est déjà trop ... Mais êtes-vous sûr de bien posséder tous les bulletins de l'AUT depuis 1950 ?.

A vos souhaits !

- Tous mes souhaits pour qu'elle grandisse très vite en renommée ... et surtout en nombre de pages. (C. Corbou).

- Sincères félicitations pour ce périodique très attendu, et qui passera très prochainement mensuel, je l'espère (Christian Triguéros).

- J'espère que votre "TRACE" suivra la trace de certaines revues américaines pleines de programmes SERIEUX, pour la gestion en général, et pas seulement de jeux, qui mettent dans les revues françaises l'ordinateur personnel au niveau du jouet (José Estrada).

- Après lecture complète du numéro 1, j'ai ressenti un grand malaise, celui d'être à nouveau rejeté car étant débutant. Presque tous les articles sont écrits pour des possesseurs de systèmes à disquettes. Or je pense que la majorité des TRS-80 vendus en France, ne possèdent que des cassettes (Jacques Laporte).

* Au sujet des revues américaines, il faut réaliser trois choses : le budget dont elles disposent ne possède aucune mesure avec le notre ; ces revues ont une expérience de plusieurs années ; les programmes qu'elles publient (sur 3 ou 400 pages, il peut effectivement y en avoir plein !) proviennent avant tout des lecteurs. Vous saisissez ce qu'il vous reste à faire ? Envoyez-nous beaucoup de programmes, et nous augmenterons la périodicité et le nombre de pages - ainsi que la publicité par la même occasion. Envoyez-nous de bons programmes sérieux et nous deviendrons une bonne revue sérieuse. C'est tout simple !
* Nous avons composé le numéro 1 de cette revue avec les articles que nous possèdions, et il se trouve que la plupart d'entre eux sont orientés vers les systèmes à disquettes, c'est vrai. Cela peut changer si vous le voulez vraiment.

Des auteurs ...

- Pourriez-vous m'indiquer si vous envisagez de publier des programmes de lecteurs, et quelles sont dans ce cas, les modalités pour vous les soumettre (Frédéric Mora).

* Oui, oui ! Nous attendons vos programmes avec impatience ! Un article manuscrit - de préférence lisible - nous conviendra parfaitement. Pour les programmes, un listing serait souhaitable lorsque cela est possible (sur papier BLANC et sans trame ...), et une cassette du programme peut nous faire gagner un temps précieux (du moins si elle est lisible elle aussi ... Deux enregistrements successifs sont conseillés). La solution luxueuse, c'est le texte (composé avec Scripsit ou Electric Pencil) et le programme sur minidisquette ... et cela passe très bien au courrier entre deux cartons.

De façon à éviter quelques problèmes, n'envoyez pas le même article à DIFFERENTES revues A LA FOIS. Votre article ne passera pas forcément dans le trimestre suivant votre envoi. Ne vous en inquiétez pas : cela peut être une question de pagination ou d'équilibre des sujets traités.
Enfin, n'exigez pas de nous une réponse systèmatique (ce n'est pas par négligence, c'est par manque de temps !).

Des questions techniques ?

- Je suis sur le point d'acheter un drive 80 pistes double faces. On m'a appris que je ne pourrai jamais lire une disquette 35 ou 40 pistes, perdant ainsi la compatibilité avec les autres utilisateurs. Qu'en est-il exactement ? J'ai entendu parler d'un doubleur réalisant cette compatibilité. Est-ce exact ? (Christian Triguéros).

* Il ne faut pas confondre la densité de pistes (obtenue mécaniquement, ainsi que la densité de faces) et la densité de secteurs (obtenue électroniquement). Les drives 80 pistes utilisent la première, ce qui signifie que 80 pistes concentriques occupent l'espace des 35 ou 40 pistes habituelles. Sur ce type de matériel, le déplacement du bras de lecture/écriture entre deux pistes est deux fois moins important que sur les unités 35 ou 40 pistes. Le mécanisme est donc plus précis et de ce fait, il est possible d'accèder deux fois plus de pistes que la normale. Pour pouvoir relire une disquette 35/40 pistes sur ces unités, il faudrait que le bras de lecture/écriture saute une piste sur deux. De l'idée à l'acte, il n'y a qu'un pas .. qui est réalisé par le système d'exploitation NEWDOS80 version 2 d'Apparat, qui autorise optionnellement la lecture des disquettes 35/40 pistes sur des unités 80 pistes. L'écriture, par contre, n'est pas conseillée par Apparat, ce qui ne vous assure donc qu'une compatibilité partielle. Dans le cas où vous visez la compatibilité totale, il est souhaitable de n'utiliser cet équipement qu'en complément à une unité 35/40 pistes.
Attention toutefois : certains constructeurs appellent "Unité 80 pistes", une unité 40 pistes réelles mais double faces ... Celles-ci sont compatibles avec les disquettes 35/40 pistes mais leur appellation est impropre. Le tout est de savoir s'il y a ou non 80 pistes PAR FACE.

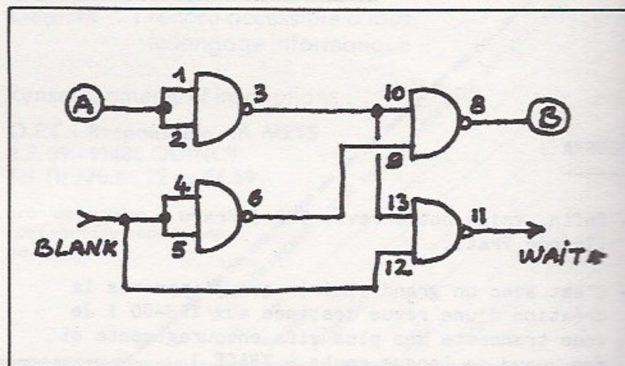
Le "doubleur" (en fait le rapport n'est que de 1,8) permet d'augmenter le nombre de secteurs dans chaque piste : 18 secteurs au lieu de 10. Pour cela, il est nécessaire d'utiliser un système d'exploitation ad hoc (NEWDOS80, LDOS, DBLDOS, ce dernier étant livré avec le doubleur). Avec le dispositif, la compatibilité est assurée puisque les disquettes simple densité continuent de fonctionner comme par le passé. Il est toujours possible de cumuler disquettes 80 pistes et doubleur. On obtient alors 80 pistes de 18 secteurs chacune.. mais le problème de compatibilité évoqué plus haut existe toujours.

- Vous décrivez dans le numéro 1, une carte permettant de doubler la capacité des unités de disquettes (Mapper 2) en utilisant des disquettes 8 pouces. Je vous signale que le "Doubler 2" permet de doubler la capacité des disquettes 5 pouces. Que pensez-vous de cette possibilité ? (Jean-Yves Bérenguer).

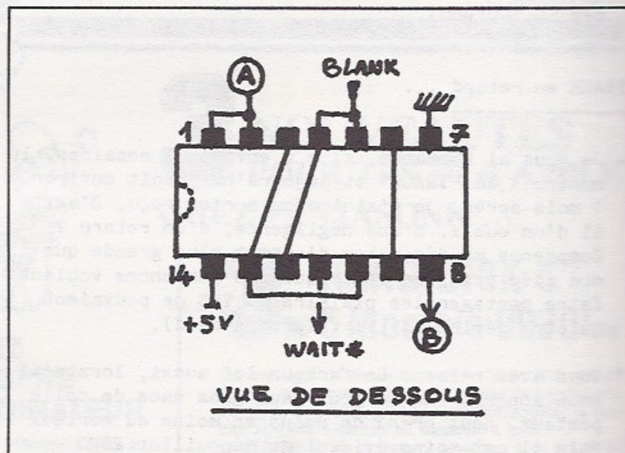
* Il est évident qu'il y a un choix à faire au départ. Le "doubler" vous permet de conserver vos disquettes d'origine et vous assure la compatibilité 5 pouces. Les disquettes 8 pouces quant à elles sont plus encombrantes et plus bruyantes. D'un autre côté, vous bénéficiez d'un support plus standard (compatible CP/M par exemple) pour une capacité plus importante (77 pistes de 17 secteurs en NEWDOS80 contre 40 pistes de 18 secteurs). De plus, la firme LNW commercialise également un "doubler" 5 et 8 pouces. Nous espèrons pouvoir vous en parler bientôt.

Une bogue ...

- Les broches 8 et 9 de Z42 utilisées pour la modification de l'affichage vidéo du numéro 1 de TRACE sont déjà occupées (selon la page 42 de "la pratique du TRS" volume 3). Peut-on les remplacer par les broches 10 et 11 ? (Jean-Claude Fabbricino).



* Honte à nous ! Il s'agit effectivement d'une erreur d'ailleurs corrigée récemment dans la dernière édition de l'ouvrage mentionné : la page 120 ne considère plus Z42 8/9 comme étant un circuit libre. Il est effectivement possible d'utiliser les sorties 10/11.

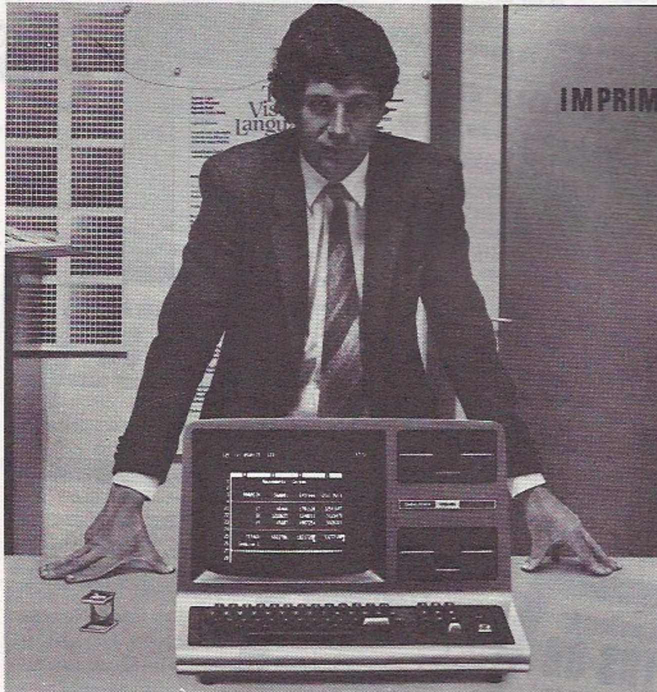


Deux montages ont été essayés par Jean-Claude Stierrer et Jean-Claude Blatry qui nous ont fait part de leur succès. Ils utilisent tous deux un circuit extérieur, l'un un 74LS132 (4 NAND 2, triggers de Schmitt), l'autre un 74LS03 (collecteurs ouverts). Le circuit peut être soit collé, soit soudé par les pattes d'alimentation 7 et 14 sur un autre circuit. Avec l'emploi du circuit à collecteurs ouverts, qui a l'avantage de permettre une utilisation multiple du signal WAIT (qui est chargé par une résistance), il nous semblerait toutefois plus prudent de prévoir des résistances de rappel sur les sorties 3, 8 et 8 (4,7K au plus 5V).

11380

Dossier n° 31 : imprimeur

"Aujourd'hui, mes offres de prix désarment mes concurrents. Et mes stocks d'encres et de papiers sont évalués au centime près. Efficace mon TRS-80 Tandy !"



«Etre imprimeur, c'est un métier difficile. Pour que mon entreprise vive, je dois offrir des prestations calculées au plus juste. C'est-à-dire gérer des stocks d'encres et de papiers qui ont des «dates de fraîcheur», anticiper des entrées de fournitures pour obtenir un marché important ou alléger les stocks, modifier le planning, profiter des taux de change, amortir les pointes imprévisibles... l'expérience et le talent ne suffisent plus.

Notre TRS-80 nous aide à gérer notre activité à la feuille de papier près et à calculer des offres de prix redoutablement compétitives. Si seulement tous mes équipements s'amortissaient aussi vite...»

Le TRS-80 de Tandy est un micro-ordinateur «professionnel» et efficace parce qu'il est parfaitement adaptable aux besoins précis de chaque utilisateur. Souple, rapide, fiable par sa conception et sa réalisation, c'est aussi un outil simple à l'usage. Il n'exige pas de formation informatique spécifique. Et toutes les opérations s'affichent automatiquement sur écran.

De larges possibilités d'extension sont prévues (imprimante, mémoire supplémentaire, etc.). Ce qui permet à l'utilisateur du micro-ordinateur TRS-80 de développer et améliorer son système selon ses besoins propres.

Tandy est le leader du marché de la micro-informatique et nos TRS-80 composent la gamme la plus performante de micro-ordinateurs. Leurs prix sont accessibles à tous les budgets. Nous vous offrons un réseau de distribution qui s'étend de jour en jour : plus de 8500 magasins dans le monde, dont quelque 600 en Europe. Notre service après-vente se charge de tous les problèmes de maintenance et de dépannage. En outre, nous travaillons en permanence au développement de matériels et logiciels toujours plus perfectionnés.

TRS-80 : une gamme d'ordinateurs à partir de 5.595 F, TVA 17,60% incluse, avec un large éventail de progiciels.

Tandy
TRS-80
The biggest name in little computers™

TC4F-M
TANDY FRANCE S.A.
Département TRS-80 - 211-213 boulevard MacDonald - 75019 PARIS
Veuillez m'envoyer une documentation détaillée sur le TRS-80
Nom
Société
Fonction
Secteur d'activité
Adresse
Code postal
Localité
Tél.

MAGAZINE

Une collaboration s'est engagée entre les éditions HACHETTE et la Société TANDY pour développer et diffuser des programmes à vocation pédagogique sur les micro-ordinateurs TRS-80. Des auteurs sont recherchés, avis aux amateurs !

La Société Pentasonic commercialise des unités de minidisquettes à intégrer soi-même, pour un prix très compétitif : 2100 F ttc (40 pistes simple-face, simple ou double densité), 2995 F (idem mais en double-face). Une alimentation est également disponible (440 F), ainsi qu'un câble de raccordement (153 F).

Cette même Société importe également le "doubleur Percom" qui, pour un prix inférieur à celui d'une unité de disquette (1995 F), permet presque de doubler la capacité des unités existantes. Le doubleur est livré sous la forme d'un petit circuit imprimé s'enfichant dans le chassis d'extension du modèle 1 (pas de soudures) et est accompagné d'un système d'exploitation (DBLDOS). Les systèmes LDOS et NEWDOS80 acceptent également ce dispositif.

La Société TRS-SOFT commercialise un utilitaire - TRSK7 - permettant aux possesseurs de "nouvelles ROM" sur modèle 1, de dupliquer certains programmes récalcitrants (EDTASM notamment). Ces derniers sont lus en mémoire, puis recopiés selon le format correct Prix : 100 F ttc.

L'Association des Utilisateurs du TRS-80 (AUT) recherche des "champions" désirant participer aux jeux les plus réputés sur TRS. Voici les scores à battre :

SUPER NOVA	104160	DECATHLON	6777
GALAXY INVASION	733160	BARRICADE	24922
METEOR MISSION 2	52920	NOVA	188940
COSMIC FIGHTER	122960	PINBALL	58920
ATTACK FORCE	59360	ALIEN	34820
SCARFMAN	31760	ROBOT ATTACK	15560

Prévenez l'AUT si vous avez fait mieux. Si vous acceptez de participer à un tournoi à Paris, indiquez les dates qui vous conviendraient et les jeux sur lesquels vous voulez concourir et adressez votre réponse à l'AUT avant le 15 septembre 1982 :

AUT, 27 Avenue Duquesne 75007 Paris

Cette association a par ailleurs annoncé le résultat du concours de programmes de jeux qu'elle a organisé fin 1981. Ces jeux, écrits par des amateurs, ont été soumis à un jury composé de huit personnes et ont été récompensés par des prix (dont trois abonnements à TRACE !). Voici les trois premiers :

1er prix : ALLIEN, par Pascal Pellier
2ème prix : SCRABBLE, par Régis Darnault
3ème prix : 6 JEUX DE CARTES, par Daniel Botton

Les juges ont tenus compte de la qualité graphique. de la valeur sonore, de la cinématique, de l'intelligence et de l'attrait présentés par chacun de ces jeux.

CP/M sur modèle 3 ? Voilà qui est désormais possible avec le "Freedom 3A" commercialisé par la firme américaine Field Engineering Consultants. De manière analogue au "mapper" d'Omikron (voir article dans ce numéro), le Freedom 3 s'enfiche sur la carte du modèle 3. La ROM et la zone clavier/écran sont repoussées en fond de mémoire tandis que la mémoire vive commence à l'adresse 0. En option (Freedom 3B), ce dispositif peut contenir des circuits RAM additionnels recouvrant la zone ROM devenue inutile. Enfin, le Freedom 3C est identique au 3B mais contient de plus, une horloge "calendrier" alimentée par batterie. Les prix sont respectivement de 199, 340 et 490 dollars selon les options.

La Société belge BCM (Bureau de Calcul Mathématique), annonce un nouveau produit : le BCM-7. Il s'agit d'une carte additionnelle pour le modèle 2 réalisant les fonctions suivantes : clavier AZERTY accentué, deux jeux de caractères à l'écran commutables par programmes (cabochons de touches modifiés) pour le français et l'allemand, un bip sonore, une horloge-calendrier en temps réel alimentée par batterie, un mode programme où les mots-clés BASIC et DOS sont affichés en une seule frappe, un mode d'émulation écran des modèles 1 et 3 (16 lignes de 64 caractères) permettant une compatibilité avec les logiciels de ces deux machines. Une adaptation accentuée du programme Scripsit est fournie à tout possesseur de la version standard. Prix : 19000 FB ou 2700 FF ttc.

Pour les utilisateurs de cassettes sur modèles 1 et 3, un "générateur de DATA BASIC" (cf. TRACE n° 1 pour une version disque) est disponible chez GRAPHIE. Avec ce programme, une zone binaire bornée par deux adresses est convertie en mémoire sous forme de lignes DATA en BASIC. Mais ce n'est pas tout ! Ce programme est accompagné d'utilitaires "system" très pratiques : "EXTBAS", qui ajoute les fonctions LINE INPUT et INSTR au BASIC niveau 2, "JKL" qui effectue une recopie d'écran sur l'imprimante, "CONCAT" pour réunir deux programmes BASIC (fonction MERGE du BASIC disque), "AUTEDT", un éditeur automatique, qui facilite les opérations d'édition d'un programme BASIC. Prix inconnu pour l'instant.
D'autre part, GRAPHIE commercialise le tout nouveau "Orchestra 90", un synthétiseur musical pour modèle 3. Basé sur le principe d'Orchestra 80 (voir Trace numéro 1), ce dispositif permet de reproduire en outre l'effet stéréophonique et les percussions. Prix : 1500 F environ.

SIVEA vient de recevoir des nouveautés sur TRS80 et Vidéo-Génie : MASTER REVERSI (jeu d'Othello très "coriace"), trois "wargames" simulant des batailles célèbres : TANKTICS, TIGERS IN THE SNOW et BATTLE OF

MAGAZINE

SHILLOH. ASYLUM est un jeu d'aventure avec graphismes en trois dimensions et SCARFMAN, une simulation du célèbre jeu électronique des cafés. Avec SPACE SHUTTLE, vous pourrez conduire la navette spaciale américaine. EDTASM-PLUS (voir notre essai dans ce numéro) complète cet arrivage d'outre atlantique dont les prix s'étendent de 150 à 395 F ttc.

Une confidence maintenant : un synthétiseur vocal à vocabulaire illimité - et de surcroit, français - sera disponible sur le modèle 1 dès le mois de mai chez Pentasonic. L'oritinalité du produit réside dans son faible coût par rapport aux produits existants (de l'ordre de 400 F), ainsi que dans le procédé utilisé : la synthèse de la voix est réalisée par programme, à partir des phonèmes de la langue française et nécessite un petit circuit imprimé connecté d'une part sur le bus TRS, d'autre part sur un haut-parleur (non-fourni). La synthèse vocale de n'importe quel texte est mise en oeuvre facilement (nous avons même pu le constater !) à partir d'un programme BASIC ou assembleur. L'ensemble est complété d'une disquette (version cassette en préparation) et d'une notice d'utilisation .

Le TRS-Couleur présente en France un grave défaut: il n'est pas compatible avec notre bon vieux système SECAM, alors même qu'il a pu être adapté en PAL pour nos amis belges et suisses. Mais la prochaine venue en force de l'ordinateur Thomson (voir notre premier numéro), ainsi que la non moins prochaine arrivée d'une version SECAM du VIC 20 de Commodore, doivent donner à réfléchir à Tandy. D'autant que les ventes du Couleurs en France doivent plutôt être freinées par le fait qu'on ne peut l'utiliser qu'en noir et blanc (un comble vu le nom du système!) ou sur un poste de télévision couleurs vraiment multistandard, ce qui n'est pas en général le cas des postes vendus sous cette dénomination. Il ne reste plus qu'à souhaiter que l'aide de spécialistes de Matra (dont une filiale fabrique des téléviseurs couleurs) ou d'ailleurs, Tandy réussisse rapidement à réaliser cette adaptation. Pour le Sicob, ou mieux encore pour la prochaine Micro-Expo...

MANIFESTATIONS:
TRACE était présent à Infora à Lyon (mi-avril), nous participerons également à Micro-Expo (14-19 juin) et à la Boutique informatique du SICOB (22 septembre-1er octobre).
Micro-Expo se tiendra au Palais des Congrès de la Porte Maillot. A noter le 17 juin, de 14 à 18 heures, une session "spéciale TRS" qui sera animée par Henri Frère, l'un des membres de notre équipe.

Si vous souhaitez utiliser le langage FORTH sur votre modèle 3, la société américaine Interactive Computer Systems devrait vous intéresser. Ell propose en effet OmniFORTH, qui nécessite une modèle 3 avec au moins 32 K octets de mémoire vive et une minidisquette, à un prix de 130$ (environ 800 FF).

La société GES vient d'annoncer la prochaine disponibilité des Génie 3 et Génie 4. Le Génie 3 est très inspiré du Modèle 3, tout en s'en éloignant suffisamment pour qu'aucun procès ne puisse lui être intenté par Tandy (comme cela avait été le cas pour le Génie 1). Le Génie 3 peut en effet fonctionner avec NEWDOS80 et un affichage de 16 lignes de 64 caractères, mais aussi sous CP/M avec un affichage de 24 lignes de 80 caractères; son processeur est un Z80 accédabt à 64 K octets de mémoire vive (extensibles selon le fabricant jusqu'à 192 K octets de MEV); les deux minidisquettes intégrées offrent chacune une capacité de 325 K octets (double piste, double densité), elles sont complétées par une interface parallèle, une interface série RS232, une horloge temps réel sauvegardée par batterie et une interface cassette(au fomat Tandy bien sûr). Le prix de ce système, qui comporte également un clavier détachable, n'est pas encore fixé mais devrait tourner aux environs de 20000 FF ttc.
Le Génie 4 quant à lui se veut une bonne alternative au TRS couleur, mais il ne sera proposé en France que lorsque le "petit" problème de l'interface avec une TV couleurs sera résolu (ah, cette SECAM!).

Adresses des sociétés mentionnées

- BCM, 24 route de la Sapinière, B 4960 Banneux, Belgique.

- Field Engineering Consultants Ltd, POB 2368, Woburn, MA 01888, USA.

- GES, 76 avenue Ledru-Rollin, 75012 Paris.

- Graphie, Galerie 92, 92 rue Saint-Lazare, 75009 Paris.

- Hachette, 24 boulevard Saint-Michel, 75005 Paris.

- Micro-Expo, SYBEX, 4 place Félix Eboué, 75583 Paris Cédex 12.

- Pentasonic, 5 rue Maurice Bourdet, 75016 Paris.

- SICOB, 6 place de Valois, 75001 Paris.

- Sivea, 31 boulevard des Batignolles, 75008 Paris.

- Tandy France, 211 boulevard MacDonald, 75019 Paris.

- TRS-Soft, 17 bis avenue de la Mésange, 94100 Saint-Maur.

coup d'œil sur...

... le LNW 80

Dans la lignée des "compatibles" modèle 1, nous avons choisi de vous parler aujourd'hui de la carte LNW, développée et commercialisée aux Etats-Unis par la société LNW RESEARCH, et importée, à notre connaissance, par au moins deux sociétés françaises (GRAPHIE et PENTASONIC).
Cette carte dont l'originalité est d'être en "kit", se présente sous la forme d'un imposant circuit imprimé (40 x 26 cm environ) double-face à trous métalisés, destiné à recevoir environ 130 circuits intégrés (disponibles séparément). Il s'agit donc d'un kit dont la réalisation n'est pas à la portée d'un débutant, d'autant que les documents fournis avec la carte se limitent aux schémas et à un manuel qui est plus une description technique de certains sous-ensembles, qu'une notice de montage "pas-à-pas" (style Heathkit par exemple). Histoire de vous "faire la main", un certain nombre de modifications sont à effectuer sur le circuit imprimé, avant toute chose (3 pages difficilement déchiffrables).
L'emplacement de tous les composants est repéré sur la carte par une sérigraphie de bonne qualité.

Description fonctionnelle

La carte LNW est composée des éléments suivants :

- Alimentations stabilisées. Les tensions logiques +5v, +12v, -12v et -5v sont créées sur la carte par une série de régulateurs intégrés (munis de radiateurs), à partir de tensions alternatives fournies par un tranformateur d'alimentation extérieur, comportant deux enroulements 9V 4A et 18V 2A (LNW qui vend aussi les ensembles tout montés à prévu un ventilateur). Un condensateur extérieur de 15000 microfarads (16v) est nécessaire.
Toutes les tensions logiques sont limitées en courant, et le +12 et le +5 sont protégées contre les surtensions. Les tensions sont ramenées sur un bornier (J2) afin d'alimenter le chassis d'extension LNW dont nous ne parlerons pas ici.

- Le microprocesseur est un Z80A (Zilog version rapide). L'horloge générale est fournie par un oscillateur à quartz de 16 MHz.

- La mémoire vive est composée de 8 boitiers 4116 (RAM dynamique, 200 ns), fournissant les 16 K octets de base (compatibilité TRS de base).

- Le rafraichissement de ces mémoires utilise le signal RFSH du Z80 (contrairement au TRS). Afin de respecter les temps d'accès des 2 RAM 2114 (mémoire écran) et des PROM BASIC, un cycle d'attente (WAIT) est ajouté lors de l'accès à ces mémoires.

- L'interprêteur BASIC en mémoire morte (6 boitiers PROM 2716) est strictement le BASIC Microsoft assurant la totale compatibilité avec le niveau 2 du TRS-80 (350 F environ, pour les six mémoires).

- L'interface cassette peut fonctionner à 500 bits/seconde (compatible modèle 1) ou à 1000 bits/seconde et présenté un contrôle automatique de gain en lecture permettant de s'affranchir des problèmes de sensibilite de réglage de volume du cassettophone.

- L'horloge temps réel fournit une interruption toutes les 25 millisecondes, tout comme le TRS.

L'oscillateur à quartz produit un signal à 16 MHz qui, après divisions, permet de disposer d'une horloge de 1,77 MHz ou 4 MHz. Cette première valeur d'horloge assure une compatibilité totale avec les logiciels du modèle 1 sur LNW (systèmes d'exploitation de disquette TRSDOS et NEWDOS, entrées/sorties cassette à 500 bauds, programmes utilisant des temps calculés par boucles d'instructions).
A l'aide d'un switch interne, il est possible de forcer en permanence, l'horloge 4 MHz. Lorsque celle-ci est utilisée, la compatibilité évoquée plus haut n'est plus assurée. Les temps de lecture/écriture disque, en particulier, ne sont plus respectés et il est nécessaire d'utiliser un système d'exploitation adéquat qui supporte ces améliorations (NEWDOS80 ou LDOS). Toutefois, le clavier LNW comporte une touche spéciale qui permet, quelle que soit la position du switch interne, de forcer l'horloge UC (Unité Centrale) à 1,77 MHz (FORCELOW).
Sur une troisième position du switch interne, l'horloge de l'unité centrale fonctionne normalement à 4 MHz mais est commutée à 1,77 MHz pendant les entrées/sorties disquette, ce qui assure la compatibilité avec les logiciels du modèle 1, tout en procurant un gain important dans la vitesse d'exécution.

Mais pour pouvoir utiliser les disquettes, il faut un chassis d'extension. Un connecteur 40 points en permet le raccordement par l'intermédiaire d'un câble plat. Le brochage de ce connecteur est identique à celui du TRS et il est donc possible d'y relier indifféremment le chassis d'interface Tandy ou celui de LNW (qui comporte en standard l'interface série V24/RS232).

Il est à noter que les synchronisations horizontales et verticales de la vidéo sont générées par un oscillateur séparé ajustable, ce qui devrait permettre sans trop de problèmes, l'utilisation de récepteurs TV européens en guise de moniteur.

La carte LNW accepte le clavier du modèle 1 ou le clavier fourni en option par LNW et qui comporte quelques touches supplémentaires (74 touches). Les caractères minuscules existent en standard sur le LNW80.

Ce qui distingue surtout le LNW80 du modèle 1 est l'adjonction d'une mémoire additionnelle (6 boitiers 4116, soit 16K x 6 bits) permettant, en liaison avec la mémoire d'écran standard (1K x 8 bits), d'augmenter les capacités d'affichage, tant pour les caractères alphanumériques et graphiques, que pour le graphisme haute résolution et la couleur, inexistants sur modèle 1. Nous allons maintenant examiner ces possibilités intéressantes.

Affichage noir et blanc

- 16 lignes de 64 caractères alphanumériques (majuscules et minuscules) et semi-graphiques (mode 0). C'est le mode compatible TRS-80. La mémoire additionnelle évoquée plus haut n'est pas utilisée.
La résolution en semi-graphique est de 128 x 48 points (coordonnées des instructions SET, RESET et POINT du BASIC). Nous utilisons l'expression "semi-graphique" car en fait, l'écran est divisé en 16 x 64 emplacements distincts. Chaque emplacement, dans ce cas, est divisé en 6 pavés et les 64 possibilités des caractères semi-graphiques correspondent aux possibilités de remplissage de ces 6 pavés (visualisées par CHR$(129) à CHR$(191)).

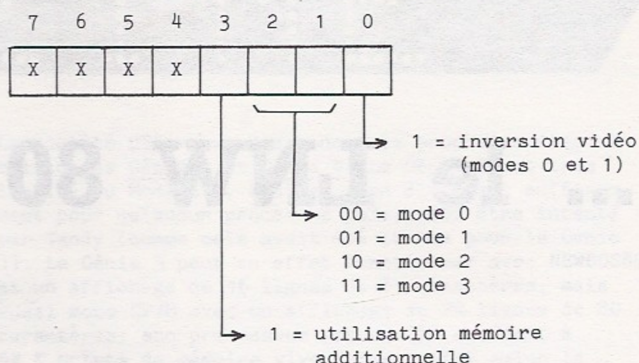
- Graphique haute résolution de 480 x 192 points (mode 1). Utilise la mémoire additionnelle uniquement. Dans ce mode, il est possible de générer par sous-programme, n'importe quel jeu de caractères. En particulier, des caractères plus petits que les caractères standards, ce qui permet un affichage de 24 lignes de 64 ou même de 80 caractères ... Quelques exemples de sous-programmes sont donnés dans la notice LNW. Ce mode offre de grandes possibilités mais n'est toutefois pas évident à mettre en oeuvre, en particulier depuis les programmes existants du modèle 1 (Scripsit, avec 24 lignes de 80 caractères, c'est le rêve, mais ce n'est pas immédiat !).
Pour les applications graphiques, c'est le mode qui offre la résolution maximale (92.160 points).
Pour chacun de ces modes, l'inversion vidéo est possible. La sortie "vidéo composite" disponible sur le connecteur "HI/RES B/W VIDEO" est compatible avec tout moniteur TV ayant une bande passante convenable.

Affichage couleur

- 128 x 192 points en 8 couleurs (mode 2). Utilise d'une part la mémoire d'écran standard pour mémoriser le caractère semi-graphique à "colorier", d'autre part la mémoire additionnelle pour contenir dans chaque pavé du caractère semi-graphique, la couleur de 4 sous-pavés (1 couleur parmi 8).
On aboutit ainsi à 128 x 192 sous-pavés de couleur.
La mémoire d'écran doit être pré-remplie de caractères semi-graphiques ou de caractères graphiques dilatés (CHR$(23)) pour obtenir un résultat correct sur un téléviseur couleur. Les caractères alphanumériques normaux (non-dilatés) donnent lieu à une mauvaise visualisation.
La sortie vers le téléviseur couleur se fait selon le standard américain NTSC, soit en vidéo composite, soit en VHF (l'emplacement du modulateur est prévu sur la carte). Une adaptation au standard SECAM impose au minimum de changer le circuit modulateur (MC1372) et la ROM de codage des couleurs (74S188) (il s'agit de possibilités entrevues par un rapide coup d'oeil sur les schémas qui n'engage pas la responsabilité de l'auteur, non spécialistes TV s'abstenir !).

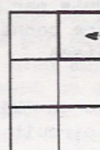
- 384 x 192 points en 8 couleurs (mode 3). Chaque point correspond à un bit dans la mémoire additionnelle (73.728 points) et peut donc être allumé (=1) ou éteint (=0).
La mémoire d'écran contient la couleur d'un pavé composé de 3 x 12 de ces points. La résolution en points est donc de 384 x 192 mais la résolution couleur n'est que de 128 x 16. La combinaison des deux doit donner de belles images !
La sortie couleur "haute résolution" se fait au standard "RVB" et pourrait peut-être, moyennant quelques modifications, être adaptée à la prise "Péritel" de nos téléviseurs français fabriqués après le 01/01/80.
Pour mémoire, aux USA un moniteur NTSC coûte environ 400 $, un moniteur RVB environ 1000 $.

Chacun des modes que nous venons de décrire est sélectionné par une instruction OUT à l'adresse 254, par l'octet suivant :



Les possibilités graphiques sont résumées ci-dessous

Mode 0

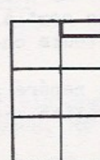


←--- accès au pavé.
128 x 48 pour tout l'écran

Mode 1

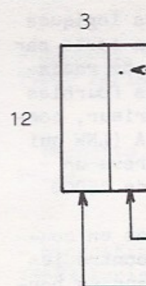
. ←--- accès au point.
480 x 192 pour tout l'écran

Mode 2



←--- accès au 1/4 de pavé (sous-pavé)
128 x 192 pour tout l'écran
1 couleur parmi 8 par sous-pavé

Mode 3



←--- accès au point
384 x 192 pour tout l'écran
1 couleur parmi 8 pour une zone composée de 3 x 12 points

couleur A

couleur B

Henri Frère

coup d'œil sur...
...le Modèle 16

Ainsi que nous le signalions très brièvement dans notre dernier numéro, Tandy a annoncé aux Etats-Unis un nouveau matériel de haut de gamme, le Modèle 16. Il s'agit en fait d'un Modèle 2 "gonflé" par l'ajout d'une carte processeur spéciale.

En effet, une carte d'unité centrale est insérée dans l'un des emplacements disponibles du châssis du Modèle 2, apportant ainsi non seulement un processeur 16 bits Motorola 68000, mais aussi 64 kilo-octets, pour une capacité totale de 128 K octets (extensible jusqu'à 512K).

Le processeur 68000 est généralement considéré comme le plus puissant des processeurs 16 bits actuels, surtout par son jeu d'instructions. Hélas, ce processeur en fait ne dispose à l'heure actuelle que de peu de logiciels d'applications, même si son logiciel de base est assez étendu: le fameux Système d'Exploitation de Disquettes (SED) Unix multi-utilisateur multitâche, compilateurs pour les langages C (langage d'écriture de systèmes), Cobol, Fortran, et bien sûr Pascal-UCSD.

Tandy annonce pour le Modèle 16 un futur SED multi-tâches, mais en l'attendant les utilisateurs en sont réduits à utiliser une version adaptée d'un SED du Modèle 2, qui permet toutefois d'effectuer des compilations Cobol sur le 68000. Car le Z-80 (8 bits) du Modèle 2 reste tujours disponible, en fait à l'heure actuelle c'est le seul des 2 processeurs pour lequel un logiciel complet existe. Tandy peut donc affirmer à juste titre que tous les programmes du modèle 2 fonctionnent sans problème avec le modèle 16!

Gageons toutefois que les sociétés de service ne manqueront pas pour développer rapidement du logiciel d'application pour le nouveau-né de Tandy: ce n'est qu'à ce moment-là que toute sa puissance pourra réellement être exploitée.

Mais voyons un peu ce qui se passe du côté des mémoires de masse. L'unité de disquettes 20 cm du modèle 2 est remplacée par une ou deux unités extra-plates de disquettes 20 cm. L'utilisation des deux faces de chaque disquette donne à chaque unité une capacité de 1.25 mégaoctet; le modèle 16 est donc commercialisé en deux versions. Toutes deux possèdent 128 K octets de mémoire vive; la première version, avec une seule unité de disquettes, coûte aux Etats-Unis 5000$ (environ 31000 FF), et la version à deux unités 5800$ (environ 35000 FF).

Sont également disponibles des disques durs de 8.4 mégaoctets, et la connexion au réseau local ARCNET que Tandy a adopté suite à ses accords avec Datapoint (qui est distribué en France, comme le monde est petit, par Matra!).

Et les possesseurs de modèle 2? Ils ne sont pas laissés à l'abandon, car ils peuvent faire transformer leur système en modèle 16. Bien sûr, leur carrosserie n'aura pas la nouvelle couleur ivoire adoptée pour le grand dernier; bien sûr, ils garderont leur épaisse unité de disquette simple face; bien sûr, ils ne pourront étendre la mémoire MEV que jusqu'à 256 K octets. Mais après tout, cela ne leur coûtera que 1500$ aux Etats-Unis, soit environ 9200 FF, tout en leur permettant d'utiliser la plupart des nouveaux logiciels (quand ils seront disponibles!).

coup d'œil sur...
..les poquettes

Exceptionnelle, cette imprimante l'est en effet: il s'agit d'une véritable mini-table traçante, capable de générer des graphiques en quatre couleurs! Voilà qui est peu banal pour un poquette (qui nécessite d'ailleurs alors une assez grosse poche!), vous en conviendrez aisément.

Côté BASIC, les nouveaux poquettes sont compatibles avec les anciens, ou plus exactement un programme qui fonctionnait sur le PC1 peut être retapé sur le PC2 en se contentant de modifications mineures: le nouveau BASIC comporte presque toutes les commandes et instructions de l'ancien, augmentées de nouvelles possibilités.

Les prix? comme dit dans notre dernier numéro, le PC2 tout seul sera commercialisé aux Etats-Unis au prix de 280$ (environ 1700 FF). A comparer avec les prix actuellement pratiqués en France pour le 1500: 2300 FF ttc, l'imprimante est elle vendue dans les 1800 FF ttc. Messieurs les importateurs et diffuseurs de matériel Sharp, un petit effort, ou bien nous irons tous acheter nos PC2 chez Tandy! (Enfin, sauf si Tandy s'aligne sur ces prix...)

Mais avec toutes ces nouveautés, que vont devenir les possesseurs de PC1 et autres PC1211? On leur avait laissé entendre qu'il y aurait beaucoup de développements et de nouveaux périphériques, il doivent à l'heure actuelle se sentir un petit peu abandonnés. (Pas dans nos colonnes en tout cas!)

En même temps que le modèle 16, Tandy annonçait un nouveau poquette, baptisé PC2. (Le poquette précédent, né Sharp PC-1211, va-t-il être rebaptisé PC1?). Les photos dont nous avions pu disposer montraient une machine plutôt différente du Sharp 1500, semblant indiquer que Tandy avait changé de fournisseur.

En fait, une lecture attentive des spécifications et de l'une et de l'autre montre qu'il s'agit en fait de la même machine, mais avec des différences de carrosserie assez importantes. En effet, alors que le clavier du 1500 est de forme trapézoïdale, un peu à la manière d'une machine à écrire, le PC2 reprend pratiquement la même disposition de touches que le PC-1211: un clavier rigoureusement rectangulaire. Par ailleurs, les affichages des deux modèles sont implantés de façon assez différente, celui du PC2 étant nettement décalé sur la droite. Tous deux sont en tout cas beaucoup plus gros que le 1211, perdant ainsi les avantages liés à celui-ci: faible poids, faible volume, faible prix, etc.

Ce qui nous avait mis la puce à l'oreille, avant même d'éplucher les caractéristiques techniques, c'était en réalité une photo très révélatrice: Tandy a diffusé des photos du PC2 installé dans "son" imprimante, laquelle ressemble comme deux gouttes d'eau à celle du 1500. Et comme celle-ci est plutôt exceptionnelle, il était donc logique de penser qu'il s'agit du même modèle d'imprimante, et donc d'ordinateur.

EDTASM +, un meilleur assembleur

EDTASM-PLUS de Microsoft
Prix : 270 à 395 F
Version cassette et disquette
Modèle 1

L'Editeur-Assembleur "cassette" le plus utilisé sur le modèle 1 est sans doute EDTASM de Tandy. Pour les utilisateurs de systèmes à disquettes il existe une version de ce programme fournie avec les différentes versions du NEWDOS.

Depuis quelque temps déjà, Microsoft commercialisait EDTASM-PLUS en version cassette et annonçait que ce programme avait été écrit par l'auteur de la version originale. Cela se traduit par un avantage indéniable pour ceux qui désirent évoluer vers ce nouveau logiciel : tous les fichiers sources, mais aussi les commandes et les pseudo-opérations de la version initiale demeurent inchangées.

Mais. le "plus" de EDTASM-PLUS se traduit en fait par des facultés supplémentaires non négligeables comme des facilités d'édition nouvelles, parmi lesquelles le déplacement de groupes de lignes, les recherches et changements globaux de certaines valeurs, l'extension automatique de toutes les lignes. Cette dernière possibilité s'avère particulièrement pratique pour ajouter des commentaires à un programme après sa mise au point.

Il y a par ailleurs une possibilité d'assembler directement en mémoire, de tester le programme, puis de retourner à l'éditeur-assembleur. On évite ainsi une perte de temps considérable, car, pendant la mise au point, il n'est plus nécessaire d'enregistrer sur support externe toutes les versions intermédiaires des programmes source et objet, ni de recharger après chaque test EDTASM, ainsi que la dernière version du programme source. Cette manière de procéder est déjà agaçante avec un système à disquettes, mais avec un système à cassettes le développement de programmes assembleur d'une certaine importance devient une tâche presque insurmontable.

EDTASM-PLUS dispose d'un programme moniteur incorporé, appelé ZBUG, pour faciliter la mise au point. Il existe même une possibilité de retour dans EDTASM sans perte du programme après un Reset (nécessaire, pour "décoincer" dans le cas où le programme test bloque la machine). Pour travailler sur un programme volumineux, il est toutefois possible d'effacer de la mémoire ZBUG, et même éventuellement l'assembleur, pour disposer de plus de place pour l'édition du programme source dans le buffer de texte.

Les améliorations d'EDTASM-PLUS ne s'arrêtent pas ici. Il faut mentionner parmi ses avantages les plus marquants. la possibilité d'utiliser des macro-instructions. des ordres d'assemblage conditionnels et des expressions d'évaluation d'adresses beaucoup plus complexes.

Le tout est accompagné d'un petit manuel et d'une carte de référence bien présentés et faciles à utiliser (en anglais malheureusement).

La version disquette de cet assembleur, sortie récemment, est encore plus complète. Le maniement des commandes de lecture, d'écriture et d'assemblage sur disquette est plus commode que celui du EDTASM modifié de NEWDOS, quoique un peu moins sûr, car EDTASM-PLUS ne demande pas la confirmation de chaque ordre donné.

La version disquette possède les pseudo-instructions supplémentaires PAGE et TITLE qui permettent d'améliorer la présentation des listes des programmes source, et une pseudo-instruction INCLUDE qui permet d'inclure au moment de l'assemblage et à partir du disque, des macro-instructions, des routines diverses ou des portions d'un programme source trop important pour tenir entièrement en mémoire. Ceci permet également de se constituer une bonne bibliothèque de marco-instructions que l'on utilisera systématiquement pour écrire ses programmes.

Cette version est livrée sur une disquette TRS-DOS avec un petit fichier contenant une série de macro-instructions d'entrées-sorties disque (OPEN, CLOSE, READ. WRITE. etc ...), un programme de patch qui permet l'adaptation du driver d'imprimante d'EDTASM à l'imprimante utilisée ainsi qu'au format du papier (nombre de caractères par ligne, taille des pages, nombre de lignes par page, caractère d'avancement de ligne).

J'ai transféré EDTASM-PLUS sur une disquette NEWDOS80-2 et je n'ai constaté à ce jour aucune anomalie.

Faut-il cependant dire que ce programme est parfait ? Non : il est tout de même un peu trop facile de perdre le buffer de texte source après un ordre (émis par erreur) de lecture d'un fichier.

Un autre petit regret, la version disquette est livrée avec le manuel cassette et un deuxième petit manuel contenant la description des commandes disquette supplémentaires. De plus la carte de référence ne contient que les informations se rapportant à la version cassette. Il aurait mieux valu un manuel refondu, et surtout une carte de référence complète.

Mais malgré ces imperfections, qui sont tout de même mineures, je n'échangerais par mon EDTASM-PLUS contre deux EDTASM de l'ancienne version !

Wolfgang Lauter

Tentez l'aventure

Ce jeu fameux écrit à l'origine en FORTRAN pour l'ordinateur PDP-10 de DEC peut maintenant tourner sur un TRS-80 équipé de 32K de mémoire et d'une disquette, laquelle comporte plus de 64K de dialogues et de descriptions ... De quoi s'amuser pendant de longues heures, mais en anglais !

La disquette "adventure" inaugure un nouveau type de jeu où le participant tient un rôle et vit des aventures dont le déroulement est fonction de ses actes. Aux Etats-Unis, on achète une "disquette" d'aventures comme nous achèterions un "livre" d'aventures. Les auteurs ne sont plus Stevenson ou Conan Doyle mais Scott Adams et Microsoft ... C'est le progrès !

Stéphane, notre jeune essayeur de 14 ans (qui a perfectionné son anglais avec ce jeu ...) a condescendu à nous livrer quelques secrets.

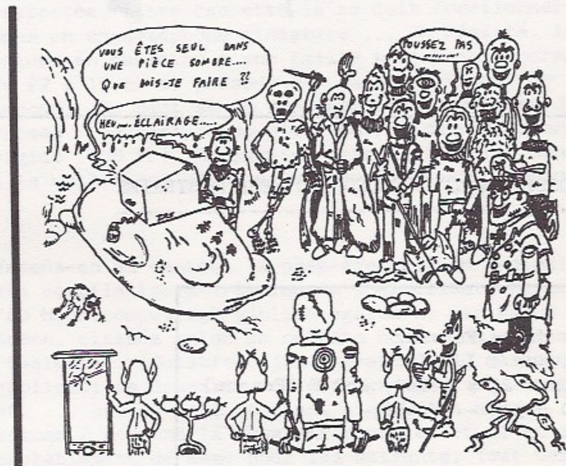
– Démoniaque ! C'est tout simplement démoniaque ! Dans ce labyrinthe de 130 pièces, il vous arrive les choses les plus horribles. Vous êtes attaqués par des animaux féroces, des dragons, des oiseaux, des pirates, etc ... durant votre recherche des 15 trésors répartis dans les endroits les plus divers de cet immense monde souterrain.
Pour nous aider, il existe 40 objets usuels parsemés de ci de là, mais attention : vous ne pouvez en porter plus de huit à la fois (hélas !), ce qui ne facilite pas la tâche.
Plusieurs passages secrets sont cachés aux détours des chemins et ils vous conduisent tout aussi bien dans la bonne que dans la mauvaise direction.
Mais tout de suite : les PETITS TRUCS ET ASTUCES.

Les trésors
----

* Une grosse pépite (nugget) est cachée dans une "pièce basse" (low room). Prendre la direction sud lorsque vous êtes au "hall des brumes" (hall of mists).

* Des oeufs d'or (golden eggs) reposent dans la chambre géante (giant room). Pour y arriver, il faut déjà parvenir à "west end two pit". Là, une plante toute rabougrie demande de l'eau. Si vous pouvez lui en donner (deux fois !) elle grandit et monte vers un trou, ce qui vous permet de grimper... pour arriver dans "narrow corridor". Prenez la direction ouest et vous y êtes !!

* Un troisième trésor (après, c'est fini ...) : la pyramide. C'est un des plus difficile à atteindre. En effet, pour y accéder, vous devez aller à "plover room" mais avant, vous devez déposer vos ustensiles pour vous faufiler dans un "boyau" étroit. Là, il y a une émeraude (encore un trésor), mais à cause du noir (vous avez dû laisser votre lampe), vous ne pouvez aller plus loin. Alors vous sortez, prenez vos objets et galoppez à "Y2" (mot magique : PLUGH) où vous prononcez le mot : PLOVER qui vous transporte par miracle dans "plover room" – cette fois avec votre équipement !! Prenez la direction nord-est et vous vous trouvez dans la pièce de la pyramide. Voilà ! Pour revenir, pas de problèmes ...
Nous vous avons dit qu'il y avait 15 trésors. La honte nous submerge ... mais n'en avons trouvé que 14 !! En voici quelques-uns : "jewellery", "rug", "silver", "coins", "chest", "diamonds", "vase", "trident", "golden chain", "rare spices" constituent le reste des richesses à trouver – et à retourner dans le building.



Phrases magiques, passages secrets, objects énigmatiques :

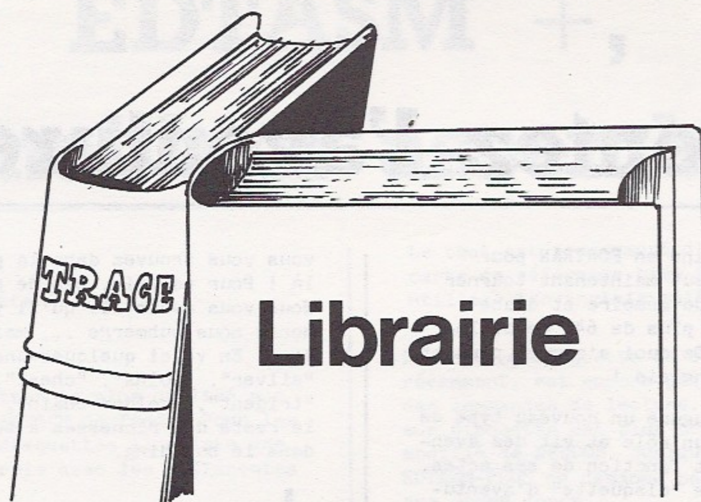
"XYZZY" écrit dans la "debris roon" sert de va-et-vient entre cette pièce et le building. "PLUGH" a la même fonction mais avec la pièce "Y2" (sert aussi à aller de la "plover room" à "Y2", mais uniquement dans ce sens). "LWPI" vous transporte de "ante room" à "software den" et vice-versa.

Pour tuer le serpent (snake) ? Facile !! Apportez le petit oiseau et libérez-le devant le monstre.
Pour vous débarrasser du dragon ... tuez-le tout simplement – et à mains nues !
A propos du vase Ming, n'essayez pas de le poser, il se casserait avec un "crash" délicat ! Donnez-le plutôt au pirate qui vous attend un peu plus loin et qui exige un don pour vous laisser la passage. Donnez à manger (food) à l'ours féroce, il deviendra gentil et vous suivra partout. Vous pourrez ainsi subtiliser la chaîne qui le retenait prisonnier. De plus il terrorisera le pirate qui cette fois, vous laissera passer sans problème !

Cette vieille porte toute rouillée refuse de vous livrer passage ? Une bonne dose d'huile fera l'affaire !

Heureusement, il est possible de sauver le jeu à un instant donné (commande SAVE), pour le reprendre le lendemain. Il faut bien se reposer un peu !

Stéphane



MICROSOFT BASIC DECODED & OTHERS MYSTERIES

Par James Farvour
Editeur : IJG Inc.
Prix : 25 $ (USA), 260 F (France)
310 pages - Broché

Ce livre est le deuxième tome d'une série qui apporte des renseignements précieux à tout possesseur d'un TRS-80. En l'occurrence, les informations données par ce livre seront particulièrement appréciées par tous ceux qui désirent programmer en assembleur sans refaire le monde, c'est à dire dans notre cas, sans réécrire entièrement les sous-programmes qui existent dans la mémoire morte (ROM) du BASIC et qui ne démandent pourtant qu'à être utilisées.

Après l'introduction et quelques explications sur la structure du BASIC niveau 2, on trouve la description de nombreuses routines d'entrée-sortie, de calcul, de conversions et de manipulation de chaînes de caractères, avec la manière de les mettre en service.

Le chapitre 3 donne des informations sur le format et la gestion des données sur cassettes et disquettes, tandis que le chapitre 4 apporte d'autres lumières sur la structure du BASIC. Notamment sur les différentes tables utilisées par lui dans la zone de communication de la mémoire vive. C'est la connaissance de cette zone qui permet la création d'extensions BASIC ainsi que la réalisation de divers drivers (programmes "pilotes") d'entrée-sortie.

Les chapitres 5 et 6 donnent des exemples d'application de ces techniques. Arrive ensuite, dans le chapitre sept, une brève explication des différences introduites par les nouvelles mémoires mortes (celles qui affichent "R/S L2 BASIC" à la mise sous tension).

Mais la partie la plus importante est en fait le chapitre 8. Celui-ci couvre à lui seul les 3/4 du livre. Il contient la liste commentée de la ROM désassemblée.

Pour des raisons de copyright cette partie du manuel n'est pas complète.

On y trouve les adresses de chaque instruction, l'instruction en HEXA, son code mnémonique, une colonne vide ou devraient se trouver les arguments des instructions (la voilà, la partie manquante !), puis les commentaires. Au dos de chaque page, figurent des commentaires plus développés, lorsque cela s'avère nécessaire.

Pour disposer d'un manuel tout à fait complet, il vous faut un désassembleur, une imprimante, des ciseaux, un baton de colle et pas mal de courage, car il faut désassembler et imprimer la ROM, puis découper et coller les parties manquantes dans les 243 pages.

Cela peut paraître fastidieux, mais alors que tout le monde ne peut pas monter un ordinateur individuel à partir d'un kit, tout le monde peut fabriquer ce "bouquin en kit". Pour mieux comprendre comment marche le Basic ou pour programmer en assembleur, ce travail pourra économiser des heures et des heures de recherche et de programmation redondante.

Wolfgang Lauter

APPLICATIONS SOFTWARE SOURCEBOOK

Radio Shack
No catalogue : 26-2114
Prix USA : 1.95 $
264 pages - Broché

Ce livre possède un caractère un peu particulier, puisqu'il s'agit de la compilation d'une liste de 1653 programmes que l'on peut trouver aux Etats-Unis sur les TRS modèles 1, 2, 3, couleur et poquette ...! Cette liste est forcément incomplète à un instant donné, mais est remise à jour périodiquement : nous avons actuellement entre les mains le volume 2 de cette série, qui, par rapport au volume 1, ne contient pas moins de 50 % de programmes supplémentaires ...Si le taux de croissance continue à ce rythme, le prochain volume risque d'être fort lourd ! Cet ouvrage de 264 pages, comporte : 34 pages d'index, dans lesquelles les programmes sont classés par application, la liste alphanumérique des programmes (par application) comportant 184 pages, et un second index de 46 pages avec classement par auteurs ou éditeurs.

Chaque page de la liste décrit 9 programmes, avec pour chacun d'eux les caractéristiques suivantes :

. nom du programme

. matériel employé : modèle, configuration mémoire

. description en une trentaine de mots

. adresse de l'auteur ou de l'éditeur

. support (cassette, disquette, listing ...)

. prix

Les applications les plus importantes sont couvertes : gestion, éducation, jeux, utilisations domestiques, industries et professions spécifiques, mathématiques.
Soulignons que ces descriptions de programmes sont dénuées de tout argument publicitaire ou commercial, pas plus que de propos critiques sur la valeur, les avantages et les inconvénients des dits programmes. Il y a d'autres ouvrages pour cela ! Malgré son caractère impersonnel, ce livre semble nous dire : "voilà ce qui existe et voici les adresses ou le trouver. Maintenant, débrouille toi !". Il fallait le faire et Tandy doit être félicité pour cette initiative.

Si vous désirez voir vos programmes figurer dans ce véritable "annuaire" des logiciels TRS, vous pouvez utiliser l'un des formulaires fournis avec le "sourcebook" et il vous en coûtera 10 $ par programme, pour une période de parution d'un an.

Alain Pinaud

ENCYCLOPEDIA FOR THE TRS-80

Edité par Wayne Green Inc.
Encyclopédie en 10 volumes
Prix : 10.95 $, broché (USA) par volume
(19.95, rigide)
100 F (France) par volume (200)

Je n'ai vu que les deux premiers ouvrages de cette collection, et pour parler franchement, ils ne m'incitent pas à acheter les suivants.

Tout ce qui peut faire l'attrait d'une encyclopédie - j'entends "de ce que l'on a coutume d'appeler une véritable encyclopédie" - ne se retrouve malheureusement pas dans cette série de volumes, à commencer par la taille. Sur la publicité (il faut toujours se méfier des publicités ...), ce qui pourrait ressembler à un gros livre (le volume 1 de la série), apparaît à côté d'une petite cassette (oh qu'elle est petite !). En admettant que l'échelle soit respectée, cette cassette-là ne doit fonctionner que dans un magnétophone miniature ... En réalité, le volume en question a une taille tout-à-fait normale, de 22 x 15 cm et possède 291 pages (274 pour le second). On peut se le procurer sous l'une des deux formes : couverture souple (100 FF) ou couverture rigide (200 FF). Ce dernier prix d'ailleurs, donne à la collection celui d'une véritable encyclopédie.

Venons-en au contenu. A première vue, on croirait à une compilation d'articles de l'excellente revue "80 Microcomputing" publiée également par Wayne Green, classés selon un certain nombre de thèmes : "business", éducation, jeux, graphiques, matériel, applications domestiques, enseignement, utilitaire, etc ... chaque article étant signé d'un auteur. Le volume 2 reprend la même philosophie, et il en est probablement de même pour les suivants. Tout cela est fort bien, mais n'est pas très nouveau : avec pas mal de publicité entre les articles, chaque volume ferait un beau "80 Microcomputing", mais pour un prix nettement inférieur. Que penseriez-vous de la démarche inverse : des articles initialement prévus pour "80 Microcomputing" qui, pour une raison quelconque, sont publiés sous forme d'ouvrages ? Bon. Admettons-le. Mais dans ce cas, ne devrait-il pas s'agir d'une sélection des meilleurs articles, ainsi groupés pour donner davantage de poids à cette encyclopédie ? Non. Cela ne peut être, à cause d'un petit mot dans la phrase précédente : il ne s'agit pas très exactement - hélas - des meilleurs articles..

Cette encyclopédie-là ne vous donnera pas le POKE nécessaire pour corriger tel ou tel problème, pas plus qu'elle ne vous apprendra à savoir comment fonctionne votre TRS-80. Vous saurez par contre, lui raccorder une télétype (?!), lui faire décoder du Morse, dessiner Snoopy, et jouer au Keno ... à moins que vous ne désiriez aborder une application "business" plus sérieuse : la gestion d'un fichier d'adresses (et un de plus !).
Après tout, Wayne Green nous a prévenu dans l'introduction : il ne s'agit pas d'une encyclopédie comme les autres ! En fait, il y a une explication : j'avais mal traduit. Il ne s'agit pas d'une encyclopédie DU TRS, mais POUR le TRS ...

Jean-Pierre du Tillet

Trucs à Broc

SI VOUS COMMENTEZ VOS DATA ...

N'employez pas le "'" habituel : cela ne marche pas sur les modèles 1 et 3. En examinant le code généré dans le "buffer" BASIC, pour le petit programme suivant :

```
10 DATA 1 , 2  ' commentaire
20 PRINT       : REM commentaire
30 PRINT       ' commentaire
```

on constate que pour la ligne 10, le code correspondant à "'" est 27H, c'est-à-dire le code de l'apostrophe. Pour la ligne 20, ": REM" donne les codes 3AH et 93H (code du ":" et code compacté de l'instruction REM).Pour la ligne 30, "'" génère les trois codes : 3AH, 93H et 0FBH. Le dernier de ces codes signifie que c'est l'apostrophe qui devra être affichée et non pas ":REM".
Pour commenter une ligne DATA, il faudra écrire :

```
10 DATA 1 , 2  :REM commentaire
ou bien :
10 DATA 1 , 2  :' commentaire.
```

Souvenez-vous toutefois que si "'" est plus rapide à écrire que ":REM", il occupe un octet de plus en mémoire !

PROTECTION MEMOIRE SUR MODELE 3.

Si vous comptez placer un programme en fond de mémoire sur modèle 3, et protéger celui-ci par un "MEMORY SIZE", sachez que l'appel du BASIC sous TRSDOS "écrase" les 48 derniers octets de la mémoire.
Tenez-en compte dans votre routine en l'implantant 48 octets plus loin ...

CLEAR ANNULE les définitions de types de variables

Bien que celà soit signalé dans le manuel du BASIC, il nous semble bon d'insister sur ce point qui peut parfois amener quelques surprises si l'on n'y prend pas garde, et qui, à première vue, n'est pas très logique. Pensez donc toujours, dans vos programmes, à placer le CLEAR avant le DEFINT (ou DEFDBL ou DEFSTR).

UN RND PAS SI ALEATOIRE QUE CELA ...

Durant la mise au point de programmes utilisant la fonction RND de génération de nombres aléatoires, il peut être intéressant de reproduire toujours les mêmes séquences de nombres. Cela peut être réalisé sur modèles 1 et 3, en forçant une valeur donnée (1 à 255) aux adresses mémoire 40AAH, 40ABH et 40ACH (16554 à 16556).
Le programme suivant, par exemple :

```
10 FOR N=16554 TO 16556 : POKE N,1 : NEXT
20 PRINT RND(100) ;
30 GOTO 20
```

délivre toujours les mêmes valeurs :
48 76 26 4 5 77 71 71 28 22 etc ...

VOUS AVEZ BIEN DIT : ERREUR DE SYNTAXE ??

En BASIC disque, on peut se demander pourquoi la ligne suivante indique systèmatiquement une erreur de syntaxe :

```
10 FOR N=&H4000 TO &H4010 : PRINT N : NEXT
```

Avez-vous trouvé ? En réalité, la syntaxe de cette ligne est tout-à-fait correcte ! La seule chose que le BASIC n'apprécie pas, c'est de trouver un espace après une valeur hexadécimale. On trouve ce défaut avec TRSDOS, LDOS et tous les systèmes qui en sont inspirés. Le TRSDOS du modèle 3 n'échappe pas à la règle. Par contre, NEWDOS80 dont toutes les extensions BASIC disque ont été ré-écrites, n'offre pas cette curiosité.

ERREUR DANS LE PROGRAMME RSM-2D

Vous connaissez probablement le programme RSM de Small System Software. C'est un moniteur en langage machine très pratique et très simple d'emploi. Il présente pourtant une anomalie : l'ordre lecture des cassettes "system" (ordre R0) a parfois du mal à s'exécuter. Cela est dû à une mauvaise valeur de constante dans la routine de lecture cassette. Voici la correction :

A l'adresse EE84 (version 48K), remplacez 85 par 76. Pour les versions 16 et 32K, les adresses sont respectivement : 6E84 et AE84.
Dans le cas où ces adresses ne correspondraient pas, cherchez la séquence :
DE FF 06 85 10 FE ... et remplacez 85 par 76.

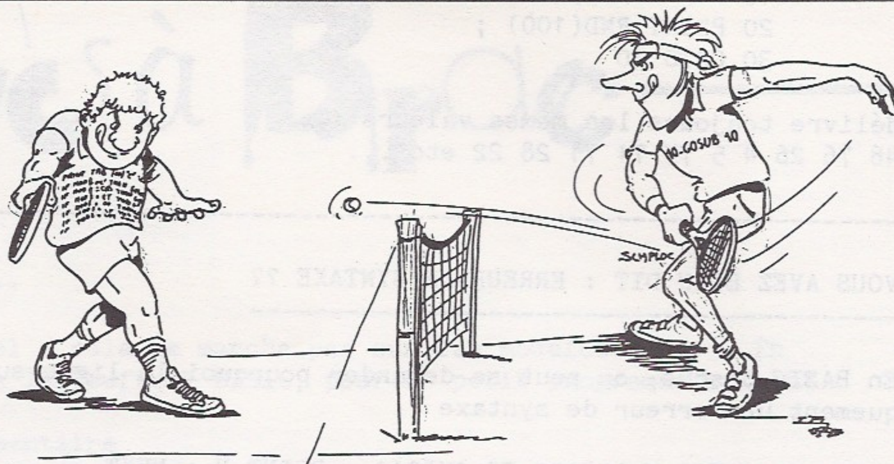
POKE OU DEFUSR ?

Vous désirez écrire un programme sur modèle 1 ou 3, qui soit suffisamment généralisé pour pouvoir tourner en version cassette (BASIC niveau 2) ou disquette (BASIC disque). Comment savoir par programme à quelle version on a affaire ?
Il suffit de tester l'octet situé à l'adresse 16396 : s'il porte la valeur 201, on est en niveau 2, pour toute autre valeur, on est en BASIC disque. Cette astuce peut être utilisée par exemple, pour définir le point d'entrée d'une routine appelée par USR (POKE en niveau 2, DEFUSR en BASIC disque) :

```
IF PEEK (16396)=201 THEN  POKE 16526,X1 : POKE 16527,X2  ELSE DEFUSR = AD
```

On trouve un autre moyen, utilisé dans le programme de contrôle de checksum décrit dans ce numéro, et qui consiste à tester l'adresse début du buffer BASIC, qui est toujours à un endroit fixe en niveau 2 (enfin ... en principe !).

Côté court



Cette rubrique contient des programmes écrits en BASIC, mais dont l'originalité est que la taille ne dépasse pas TROIS lignes au maximum, chacun d'eux effectuant une fonction bien précise. Creusez-vous les méninges et soumettez-nous les vôtres !

- ROUTAGES IMPRIMANTE / ECRAN : 1 LIGNE

```
10 FOR AD=16414 TO 16420 : POKE AD+8,PEEK(AD) : NEXT
```

Il vous arrive probablement de lancer un programme contenant les ordres imprimante : LLIST ou LPRINT. Si votre imprimante n'est pas raccordée, cela peut durer fort longtemps ...
Lancez ce petit programme et tout ce qui s'adresse habituellement à l'imprimante sera affiché sur l'écran. Attention toutefois : certains programmes gèrent eux-mêmes leurs impressions et ne passent pas par "l'entrée principale" du driver imprimante...

Pour diriger l'écran sur l'imprimante, c'est aussi simple. Il suffit de lancer :

```
10 FOR AD=16422 TO 16428 : POKE AD-8,PEEK(AD) : NEXT
```

Par contre, revenir à l'état initial n'est pas possible si l'on a pas pris soin de sauver les anciens contenus des adresses "pokées".

Une variante consiste à "croiser" l'écran et l'imprimante :

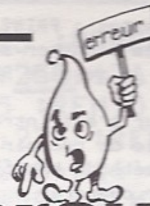
```
10 FOR AD=16422 TO 16428 : A=PEEK(AD-8) : POKE AD-8,PEEK(AD) : POKE AD,A : NEXT
```

Cette fois-ci, le retour à l'état initial est possible : il suffit de relancer le programme une seconde fois !

- TRIANGLES RECTANGLES : 1 LIGNE

```
10 X=RND(127) : Y=RND(47) : SET (X OR Y,Y) : GOTO 10
```

Les nombres nous réservent toujours quelques surprises ! Ainsi, qui pourrait penser que ce programme dont les données semblent parfaitement aléatoires, dessine un joli motif composé de triangles rectangles de différentes tailles ?
Il suffit d'attendre un instant et vous les verrez apparaître ...



supprimer les ereurs de nos programmes

Il est bien rare, en recopiant la liste d'un programme publié dans une revue, (TRACE comme les autres ... mais pour peu de temps !), de ne pas introduire une ou deux erreurs difficiles à dépister par la suite et risquant de compromettre l'exécution du programme initialement prévue : la lettre "O" est remplaçée par le chiffre "0" par exemple, ou une lettre est frappée en utilisant la touche SHIFT, pour ne citer que les plus courantes. Parfois, certaines erreurs sont tellement vicieuses que l'on peut passer dix fois dessus sans pour autant les détecter.

Eh bien à partir d'aujourd'hui, ce genre de problème ne se rencontrera plus ! C'est une bonne nouvelle, non ?

Dorénavant, tous les programmes BASIC que nous vous proposerons dans cette revue, seront pourvus d'un contrôle de "checksum" situé à la fin de chaque ligne, sous la forme d'un commentaire. Ce checksum est généré par un programme spécial : GENCS, dont il ne sera pas question ici.

Qu'est-ce qu'un "checksum" ? La traduction de l'anglais donne : somme de contrôle. Il s'agit d'un procédé largement utilisé en informatique, pour contrôler l'intégrité d'un bloc d'informations. Il existe plusieurs méthodes de contrôle et celle que nous avons choisi est une des plus simples : en additionnant tous les octets d'un bloc d'informations modulo 256 (c'est-à-dire qu'après 255, on repasse à zéro) et en y ajoutant l'octet de checksum, on doit trouver la valeur zéro. Si tel n'est pas le cas, c'est qu'une erreur a été introduite quelque part.

Exemple : le checksum des quatre lettres ABCD est 'F6'. Pourquoi ?
Le code ASCII (hexa) de ces quatre lettres est 41, 42, 43 et 44. En additionnant ces quatre valeurs (en hexa), on obtient '10A'. Modulo 256, il reste '0A'. En ajoutant le checksum 'F6', cela fait '100', c'est-à-dire '00' si l'on ne conserve que l'octet de poids faibles.

Pour tous les programmes BASIC publiés, une valeur de checksum sera donnée ligne par ligne. Il vous suffira donc d'introduire chaque ligne de programme telle qu'elle figure dans le listing, y compris le checksum qui y est mentionné. L'exécution du programme utilitaire donné plus loin, se chargera de lister les lignes sur lesquelles vous avez commis des erreurs.

Pour faciliter les choses, les ESPACES et les COMMENTAIRES ne participent pas au calcul du checksum, ce qui n'est pas le cas des numéros de lignes, qui ne doivent pas être modifiés. En conséquence, il n'est pas conseillé de renuméroter le programme AVANT sa vérification.
Les lignes suivantes sont identiques du point de vue du checksum :

```
10 CLS:PRINTA'ZONE COMMENTAIRE 45
10 CLS : PRINT A    : REM ZONE COMMENTAIRE 45
10 CLS : PRINT A ' 45
```

et possèdent toutes trois un checksum de 45. Tout changement dans la ligne sera détecté :

```
20 CLS : PRINT A    --- donnerait un checksum de 3B
10 CLS : PRIMT A    --- donnerait un checksum de 6B
10 CLS   PRINT A    --- donnerait un checksum de 7F
```

Les commentaires peuvent être introduits sous les deux formes ":REM" ou "'", toutefois, il est nécessaire d'utiliser la première forme en ce qui concerne les instructions DATA (voir à ce sujet : Trucs à brac dans ce numéro).

Programme de vérification de checksum : VERCS

Le programme VERCS écrit à l'origine en langage machine (pour des raisons de performances), est donné sous la forme d'instructions DATA contenues dans un programme BASIC. Néanmoins, un listage source est donné pour ceux qui possèdent l'assembleur EDTASM. La présente version qui se charge en fond de mémoire, est prévue pour une configuration de 48K et tourne sur les TRS-80 modèles 1 et 3. Pour les versions 16 ou 32K, quelques modifications seront à prévoir. De plus, avant de charger le programme BASIC à contrôler, il est nécessaire de protéger VERCS à l'aide d'un 'MEMORY SIZE' adéquat.

Configuration mémoire	Réponse à Mémory size
16K	32464
32K	48848
48K	65232

Marche à suivre

- Au niveau BASIC, répondre au 'Memory size' selon l'une des valeurs données dans le tableau ci-dessus et correspondant à votre configuration mémoire.

- Entrez le programme BASIC en ignorant les lignes repérées par un '*', sauf si votre version est en 48K.

- Le cas échéant, complétez le programme par le groupe de 11 lignes correspondant à votre version (16 ou 32K).

- Sauvez le programme avant de le lancer ...

- Faire RUN

Si tout se passe bien, frappez : PRINT USR(0)
Le programme que vous venez d'entrer se contrôle alors lui-même (... !) et le message READY doit apparaître.
Bien entendu, si une erreur s'est introduite à ce niveau-là, il est difficile de prévoir la suite des évênements ... Suivez la méthode habituelle de mise au point en commençant par comparer soigneusement votre programme avec le listing donné.

En admettant que ça marche (ça doit marcher !), essayez d'ajouter une ligne :

5 CLS

et frappez PRINT USR(0)

Le programme doit afficher :

- LIGNE : 00005, CHECKSUM TROUVE : 77

Refrappez alors la ligne 5 en y ajoutant cette valeur :

5 CLS ' 77

et frappez de nouveau PRINT USR(0)

Si seul le message READY apparaît, votre programme a surement été recopié correctement. Félicitations !

Alain Pinaud

```
 10 '                                                    '  F6
 20 ' programme de verification de checksum d'un         '  EC
 30 ' programme BASIC charge en memoire                  '  E2
 40 ' VERCS - Copyright (C) TRACE 1982                   '  D8
 50 '                                                    '  CE
*60 CM=48        ' 16, 32 ou 48 selon config memoire     '  F3
 70 IF CM=48 THEN AD=-304 ELSE IF CM=32 THEN AD=-16688 ELSE I
F CM=16 THEN AD=32464 ELSE PRINT "ERREUR CONFIG MEMOIRE":STO
P:REM2D
 80 '   B0
 90 '   Chargement de la routine         '   A6
 100 '   9C
 110 FOR N=AD TO AD+252:REMC1
 120 READ A : POKE N,A:REM16
 130 NEXT:REMF7
 140 '   74
 150 '    Initialisation VERCS selon la         '   6A
 160 '    version (cassette ou disquette)       '   60
 170 '   56
 180 IF AD<0 THEN K=65536+AD ELSE K=AD:REM79
 190 K2=INT(K/256) : K1=K-(K2*256):REM31
 200 IF PEEK(16549) > 67 THEN DEFUSR=AD ELSE POKE 16526,K1 :
POKE 16527,K2:REMCF
 210 CLS:REMAA
 220 PRINT "POUR EXECUTER VERCS, FRAPPER : PRINT USR(0)":REM8
2
 230 '   1A
 240 '   Routine en code machine      '   10
 250 '   06
 1000 DATA 221,42,164,64,14,0,221,229,225,221 :REM0C
 1010 DATA 126,0,221,182,1,202,25,26,221,126 :REM3C
 1020 DATA 2,221,134,3,129,79,17,4,0,221 :REMEE
 1030 DATA 25,221,126,0,183,40,31,254,32,40 :REM55
 1040 DATA 23,254,147,32,14,121,214,58,79,221 :REMD5
 1050 DATA 35,221,126,0,183,32,248,24,9,221 :REM35
 1060 DATA 126,0,129,79,221,35,24,219,221,126 :REMC2
*1070 DATA 254,81,205,106,255,7,7,7,7,71 :REMA8
*1080 DATA 221,126,255,205,106,255,176,129,221,35 :REME5
*1090 DATA 40,168,122,237,68,71,205,114,255,50 :REM6E
*1100 DATA 202,255,120,15,15,15,15,205,114,255 :REM78
```

```
* 1110 DATA 50,201,255,35,35,94,35,86,33,175 :REME8
* 1120 DATA 255,205,123,255,221,229,33,166,255,126 :REMB9
  1130 DATA 254,1,40,15,229,205,51,0,225,35 :REM1D
  1140 DATA 58,128,56,203,71,32,249,24,236,221 :REM6A
* 1150 DATA 225,195,212,254,214,48,254,10,248,214 :REMCE
  1160 DATA 7,201,230,15,198,144,39,206,64,39 :REM86
* 1170 DATA 201,1,240,216,205,150,255,1,24,252 :REM6A
* 1180 DATA 205,150,255,1,156,255,205,150,255,1 :REM2D
* 1190 DATA 246,255,205,150,255,1,255,255,62,47 :REM05
  1200 DATA 229,213,225,60,9,56,252,237,66,235 :REM27
  1210 DATA 225,119,35,201,45,76,73,71,78,69 :REM79
  1220 DATA 32,58,32,0,0,0,0,0,32,44 :REM30
  1230 DATA 32,67,72,69,67,75,83,85,77,32 :REMEC
  1240 DATA 84,82,79,85,86,69,32,58,32,0 :REM19
  1250 DATA 0,13,1:REM75
```

Adaptation 16K

```
60 CM=16        ' 16, 32 ou 48 selon config memoire '   F8
1070 DATA 254,81,205,106,127,7,7,7,7,71 :REMAA
1080 DATA 221,126,255,205,106,127,176,129,221,35 :REME7
1090 DATA 40,168,122,237,68,71,205,114,127,50 :REM70
1100 DATA 202,127,120,15,15,15,15,205,114,127 :REM7C
1110 DATA 50,201,127,35,35,94,35,86,33,175 :REMEA
1120 DATA 127,205,123,127,221,229,33,166,127,126 :REMBF
1150 DATA 225,195,212,126,214,48,254,10,248,214 :REMD0
1170 DATA 201,1,240,216,205,150,127,1,24,252 :REM6C
1180 DATA 205,150,127,1,156,255,205,150,127,1 :REM24
1190 DATA 246,255,205,150,127,1,255,255,62,47 :REM07
```



Adaptation 32K

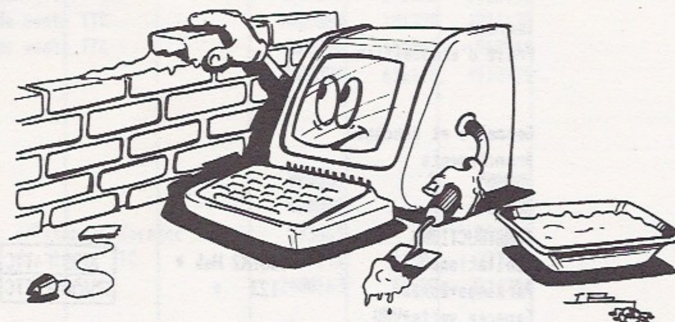
```
60 CM=32        ' 16, 32 ou 48 selon config memoire '   FA
1070 DATA 254,81,205,106,191,7,7,7,7,71 :REMA9
1080 DATA 221,126,255,205,106,191,176,129,221,35 :REME6
1090 DATA 40,168,122,237,68,71,205,114,191,50 :REM6F
1100 DATA 202,191,120,15,15,15,15,205,114,191 :REM7A
1110 DATA 50,201,191,35,35,94,35,86,33,175 :REME9
1120 DATA 191,205,123,191,221,229,33,166,191,126 :REMBC
1150 DATA 225,195,212,190,214,48,254,10,248,214 :REMCF
1170 DATA 201,1,240,216,205,150,191,1,24,252 :REM6B
1180 DATA 205,150,191,1,156,255,205,150,191,1 :REM22
1190 DATA 246,255,205,150,191,1,255,255,62,47 :REM06
```

```
       : PROGRAMME DE CONTROLE DE CHECKSUM D'UN
       : PROGRAMME BASIC CHARGE EN MEMOIRE
       : COPYRIGHT (C) 1982 - TRACE
         ORG     0FED0H      ; SELON CONFIG MEMOIRE
BASIC    EQU     1A19H       ; POINT REPRISE BASIC
CS       LD      IX,(40A4H)  ; DEBUT BASIC
NEXTL    LD      C,0         ; INIT VALEUR CS
         PUSH    IX          ; SAUVE AD DEBUT LIGNE
         POP     HL          ; DANS HL
         LD      A,(IX)      ;
         OR      (IX+1)      ; AD CHAINE
         JP      Z,BASIC     ; FIN DU PROGRAMME
         LD      A,(IX+2)    ;
         ADD     A,(IX+3)    ;
         ADD     A,C         ;
         LD      C,A         ; CUMUL CS
         LD      DE,4        ;
         ADD     IX,DE       ; +4 --> IX
LIGN     LD      A,(IX)      ;
         OR      A           ; FIN LIGNE?
         JR      Z,FLIGN     ; OUI
         CP      20H         ; ESPACE?
         JR      Z,BLANC     ; OUI
         CP      93H         ; REM?
         JR      NZ,N2P      ; NON
         LD      A,C         ; LE ":" EST RETIRE
         SUB     3AH         ; DU CHECKSUM
         LD      C,A         ;
CHERCH   INC     IX          ; CHERCHE FIN LIGNE
         LD      A,(IX)      ;
```

```
00310          OR     A             ;
00320          JR     NZ,CHERCH     ;
00330          JR     FLIGN         ;
00340 N2P      LD     A,(IX)        ;
00350          ADD    A,C           ;
00360          LD     C,A           ; CUMUL CS
00370 BLANC    INC    IX            ; EVOLUTION POINTEUR
00380          JR     LIGN          ; ET SUITE
00390 FLIGN    LD     A,(IX-2)      ;
00400          LD     D,C           ; SAUVE CS PARTIEL
00410          CALL   ASCBIN        ; CONVERSION BIN
00420          RLCA                 ;
00430          RLCA                 ;
00440          RLCA                 ;
00450          RLCA                 ;
00460          LD     B,A           ; SAUVE QUARTET FORT
00470          LD     A,(IX-1)      ;
00480          CALL   ASCBIN        ; CONVERSION BIN
00490          OR     B             ; CS EN BINAIRE
00500          ADD    A,C           ;
00510          INC    IX            ;
00520          JR     Z,NEXTL       ; OK. LIGNE SUIVANTE
00530 ; LE CHECKSUM CORRECT N'EST PAS TROUVE
00540          LD     A,D           ; CS PARTIEL
00550          NEG                  ;
00560          LD     B,A           ; SAVE CS BINAIRE
00570          CALL   BINASC        ; QUARTET FAIBLE
00580          LD     (A2CS),A      ;
00590          LD     A,B           ;
00600          RRCA                 ;
00610          RRCA                 ;
00620          RRCA                 ;
00630          RRCA                 ;
00640          CALL   BINASC        ; QUARTET FORT
00650          LD     (A1CS),A      ;
00660 ; RECHERCHE NUMERO DE LIGNE POUR EDITION
00670          INC    HL            ;
00680          INC    HL            ;
00690          LD     E,(HL)        ;
00700          INC    HL            ;
00710          LD     D,(HL)        ;
00640          CALL   BINASC        ; QUARTET FORT
00650          LD     (A1CS),A      ;
00660 ; RECHERCHE NUMERO DE LIGNE POUR EDITION
00670          INC    HL            ;
00680          INC    HL            ;
00690          LD     E,(HL)        ;
00700          INC    HL            ;
00710          LD     D,(HL)        ;
00720 ; CONVERSION NO DE LIGNE EN DECIMAL
00730          LD     HL,BUFLIG     ; BUFFER ASCII
00740          CALL   BINDC         ;
00750          PUSH   IX            ; SAUVE POINTEUR LIGNE
00760 ; EDITION MESSAGE
00770          LD     HL,ERREUR     ;
00780 SUITE    LD     A,(HL)        ;
00790          CP     1             ; FIN TEXTE?
00800          JR     Z,FEDIT       ;
00810          PUSH   HL            ;
00820          CALL   33H           ; AFFICHAGE 1 CARAC
00830          POP    HL            ;
00840          INC    HL            ;
00850 ; GEL DU LISTAGE SI SHIFT EST ENFONCEE
00860 ATT      LD     A,(3880H)     ;
00870          BIT    0,A           ; SHIFT ENFONCE?
00880          JR     NZ,ATT        ; OUI
00890          JR     SUITE         ;
00900 ; SUITE DU TRAITEMENT
00910 FEDIT    POP    IX            ; RESTAURE PTR LIGNE
00920          JP     NEXTL         ; LIGNE SUIVANTE
00930 ; ROUTINE CONVERSION HEXA ASCII --> BINAIRE
00940 ASCBIN   SUB    '0'
00950          CP     10
00960          RET    M
00970          SUB    7
00980          RET
00990 ; ROUTINE CONVERSION BINAIRE --> HEXA ASCII
01000 BINASC   AND    0FH
01010          ADD    A,90H
01020          DAA
01030          ADC    A,40H
01040          DAA
01050          RET
01060 ; ROUTINE CONVERSION BINAIRE --> DECIMAL
01070 BINDC    LD     BC,-10000
01080          CALL   CBD
01090          LD     BC,-1000
01100          CALL   CBD
01110          LD     BC,-100
01120          CALL   CBD
01130          LD     BC,-10
01140          CALL   CBD
01150          LD     BC,-1
01160 CBD      LD     A,'0'-1
01170          PUSH   HL
01180          PUSH   DE
01190          POP    HL
01200 CBD1     INC    A
01210          ADD    HL,BC
01220          JR     C,CBD1
01230          SBC    HL,BC
01240          EX     DE,HL
01250          POP    HL
01260          LD     (HL),A
01270          INC    HL
01280          RET
01290 ;
01300 ; BUFFER MESSAGE
01310 ERREUR   DEFM   '-LIGNE : '
01320 BUFLIG   DEFS   5
01330          DEFM   ' , CHECKSUM TROUVE : '
01340 A1CS     DEFS   1
01350 A2CS     DEFS   1
01360          DEFB   0DH
01370          DEFB   1
01380          DEFS   48        ; POUR MODELE 3
01390 ZZ       EQU    $
01400          END
```

OUF! OOo°

Décidez à l'aide de Visicalc



Visicalc est disponible actuellement sur les trois modèles de "table" de TRS-80.
La version la moins coûteuse est celle du modèle 1 qui vaut moins de 600 F !
Il faut impérativement 48K en modèle 1 et 3, et 64K en modèle 2 sinon les tableaux de calculs réalisables sont ridiculeusement petits.
Le génie des ingénieurs américains qui ont conçu ce logiciel est d'avoir passé du temps à observer ce que font les comptables, les analystes financiers, les ingénieurs, les contrôleurs de gestion avec leurs tableaux de calculs réalisés avec un crayon, une gomme et une calculette.

Le domaine idéal d'application peut se définir ainsi "Tout tableau de calcul qui sert à répondre à la question : que se passerait-il si ... ?".

Ce tableau de calcul pourra être conservé tel quel sur disque, imprimé au format désiré par l'utilisateur, etc ...

Les applications pratiques sont donc innombrables : feuilles de calcul de prix de revient, feuilles de calculs techniques, budgets, tableaux d'analyse financière, bilan et compte d'exploitation simplifiés, planning de trésorerie, calculs d'amortissements, recherche de point d'équilibre, besoins en fonds de roulement, suivi de travaux mensuels, etc.. Sans oublier la tenue du budget familial et le calcul des impôts !!

Prenons l'exemple d'un budget de construction d'un immeuble d'habitation.
Le constructeur se trouve en face de quatre inconnues :

- le prix de vente
- le prix d'achat du terrain
- le coût de construction
- la marge qui rémunèra les capitaux investis

Au départ de l'opération il connaît :

. le prix de vente du terrain "demandé"
. le coût de construction moyen de l'époque
. les prix de vente du monde immobilier
. la marge en dessous de laquelle il ne trouvera par les capitaux nécessaires pour réaliser l'opération.

Le constructeur établit donc un budget prévisionnel où 4 paramètres sont inconnus mais bornés (au sens mathématique du terme).
Pour faire le calcul de ce budget à la main, il faut à peu près une demi-heure et pour bien le présenter une autre demi-heure.

L'expérience prouve que pour aboutir à une solution viable, il faut souvent recommencer ce budget vingt fois en le tirant par tous les bouts !

Avec Visicalc, on fait un nouveau budget toutes les minutes si cela est nécessaire !

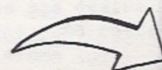
Comment s'y prendre pratiquement ?
Il faut établir un document de base qu'on appelle généralement matrice et qui est le tableau "en blanc" avec toutes ses écritures littérales et ses formules de calcul.
Cette matrice peut être obtenue en "blanchissant" un premier tableau qui aura servi de brouillon.

Le choix du format des colonnes dépend ici du nombre de chiffres à faire apparaître (9 dans le cas présent).

Le tableau présenté nécessite 9 colonnes de 10 caractères, il faut en tenir compte pour loger les libellés et les variables auxiliaires dans des cases indépendantes permettant de les utiliser dans les calculs.

Pour la matrice, il faut supprimer l'ordre de calcul automatique ce qui permettra d'introduire les données plus rapidement ultérieurement.

Finalement le document pourra être mis pratiquement sous la forme où il était utilisé par les calculs manuels avec édition par une dactylo.
Mais en plus, et sans imprimer, une variété presque infinie de simulations pourra être faite à l'écran avec la possibilité de stocker sur disque les solutions intéressantes.
Un outil d'une très grande souplesse et dont les applications semblent illimitées.



J.P. STREMLER

S.C.I. LA RESIDENCE
DES ROSES
LA SOLIDARITE

BUDGET DE CONSTRUCTION
****
VALEUR: MAI 1981

Le:10 MAI 1981

			PRIX HORS TVA	TVA SUR REGLEMENTS	PRIX TTC
RECETTES					
VENTE A LA CAISSE DE RETRAITE			35029954	6165272	41195226
TOTAL DES RECETTES			35029954	6165272	41195226
DEPENSES					
Terrain			4545420	559996	5105416
Frais d'acquisition			76600	0	76600
			4622020	559996	5182016
Geometre et sondages			55272	9728	65000
Branchements			365646	64954	430000
			420918	74682	495000
CONSTRUCTIONS					
Habitations	6887M2 Hab *	3285 F TTC	19237920	3385875	22623795
Parkings+Boxes	123 *	25650 F TTC	2682780	472170	3154950
Espaces verts+VRD			1443771	254104	1697875
			23364471	4112149	27476620
PRESTATIONS DE SERVICES					
Architecte	6% ht des travaux TTC		1648597	290153	1938750
Gestion :	2.5% ht du Prix de vente TTC		1029880	181258	1211138
Vente :	2% ht du Prix de vente TTC		823904	145007	968911
			3502381	616418	4118799
FRAIS ANNEXES					
Frais financiers			700000	0	700000
TAXES et Divers			950000	0	950000
Assurance M.O.	2%ttc s/Tx+Archi+Socotec		593865	0	593865
Socotec :	.86% ht des travx TTC		236298	41588	277886
			2480163	41588	2521751
TOTAL BRUT			34389953	5404833	39794186
T.V.A.complementaire	(6165272 -5404833)				760439
PRIX DE REVIENT ttc					40554625
MARGE:	1.5% P.V. ttc				640601
TOTAL					41195226

S.C.I. LA RESIDENCE
DES ROSES
LA SOLIDARITE

BUDGET DE CONSTRUCTION
****
VALEUR: MAI 1981

Le:10 MAI 1981

	PRIX HORS TVA	TVA SUR REGLEMENTS	PRIX TTC
RECETTES			
VENTE A LA CAISSE DE RETRAITE	35029954	6165272	41195226
TOTAL DES RECETTES	35029954	6165272	41195226
DEPENSES			
Terrain	4545420	559996	5105416
Frais d'acquisition	76600	0	76600
	4622020	559996	5182016

Geometre et sondages		55272	9728	65000
Branchements		365646	64954	430000
		420918	74682	495000
CONSTRUCTIONS				
Habitations	6887M2 Hab * 3285 F TTC	19237920	3385875	22623795
Parkings+Boxes	123 * 25650 F TTC	2682780	472170	3154950
Espaces verts+VRD		1443771	254104	1697875
		23364471	4112149	27476620
PRESTATIONS DE SERVICES				
Architecte	6% ht des travaux TTC	1648597	290153	1938750
Gestion :	2.5% ht du Prix de vente TTC	1029880	181258	1211138
Vente :	2% ht du Prix de vente TTC	823904	145007	968911
		3502381	616418	4118799
FRAIS ANNEXES				
Frais financiers		700000	0	700000
TAXES et Divers		950000	0	950000
Assurance M.O.	2%ttc s/Tx+Archi+Socotec	593865	0	593865
Socotec :	.86% ht des travx TTC	236298	41588	277886
		2480163	41588	2521751
	TOTAL BRUT	34389953	5404833	39794186
T.V.A.complementaire	(6165272 -5404833)			760439
	PRIX DE REVIENT ttc			40554625
	MARGE: 1.5% P.V. ttc			640601
	TOTAL			41195226

professionnel

Les TRS sont sympas, ils calculent pour vous

Le TRS-80 au service des PME ? Voici un exemple qui démontre clairement que cette petite machine n'est pas seulement le jouet des amateurs d'informatique que, mais peut se révéler être un instrument qui apporte aujourd'hui sa contribution à la bonne marche des entreprises.

Depuis quelques années, une société de services de Garonor dont la vocation s'orientait de manière privilégiée vers la résolution des problèmes de transport, a mis au point un système à base de TRS-80 qui permet d'informatiser et d'automatiser le transport de colis à travers toute le France.

Les expéditeurs, répartis en différents points de la métropole, ont à leur disposition un TRS-80 modèle 1 (plus rarement modèle 2). Cet ordinateur de bas prix leur permet de réaliser avec leur propre personnel, sans formation particulière, et au fur et à mesure des besoins, la constitution des fichiers, l'édition automatique des étiquettes des colis aux noms et adresses de leurs clients, la création des récépissés de transport et des bons de remise de la marchandise, et enfin, l'édition des relevés des expéditions et le contrôle des envois.

Dans certains sites, plus besoin n'est de lire le poids d'un colis et d'en déterminer l'affranchissement : le TRS est relié à une bascule qui lui transfère l'information nécessaire à ce traitement.

Cette application est opérationnelle depuis déjà quelques mois, et est utilisée quotidiennement avec succès. Elle contrôle entièrement l'acheminement des colis, depuis leur point de départ (expéditeurs clients de l'entreprise de transport) jusqu'à Garonor où les camions les conduiront vers leurs destinations finales.

Différents niveaux de services peuvent être envisagés pour l'expéditeur, en fonction de ses désirs, de ses besoins ou de ses moyens, ce qui rend cette application très modulaire. Nous parlerons ici d'une configuration se situant dans la moyenne. Mais ... transportons-nous sur le site !

AU DEPART ...

Nous nous trouvons actuellement dans une ville de province où se trouve précisément un client important de cette société de transport. Une opératrice est occupée à faire de la saisie sur le clavier d'un modèle 1. Celui-ci est relié à une imprimante parallèle et à une bascule électronique.

Un colis vient d'être posé sur la bascule qui oscille encore, tandis que l'opératrice s'applique à entrer les informations qui lui sont demandées concernant cette expédition : destinataire, nombre de colis, type de port, message éventuel, etc ...

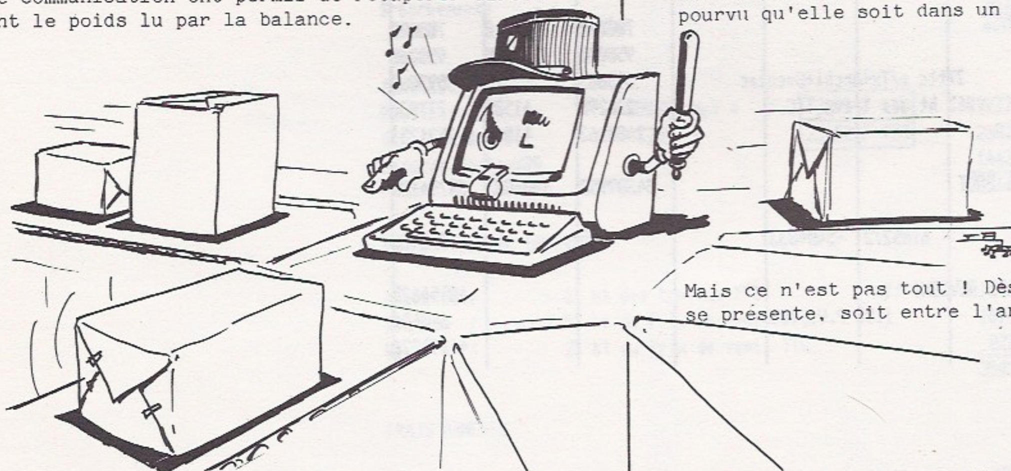
Lorsqu'elle en arrive à la question "poids du colis ?", ... oh miracle ! : le poids s'affiche automatiquement sur l'écran ! Une liaison "télécommunication" via la carte série de l'ordinateur et un module de communication ont permis de récupérer directement le poids lu par la balance.

Les risques d'erreur étaient importants et aucun contrôle efficace ne pouvait être effectué sur le passage des colis qui ne restaient parfois que quelques minutes dans cet entrepôt.

Des ordinateurs ont été mis en place près des goulottes de débarquement, le long des tapis roulants sur lesquels passent les colis qui viennent d'être déchargés.

Un "scanner" (lecteur optique) à rayon laser lit le "code à barres" situé sur l'étiquette du colis, et l'envoie vers un ordinateur. Dans celui-ci, un logiciel spécial interprète cette information et transmet à la chaîne de tri les ordres nécessaires qui vont provoquer le bon acheminement du colis vers la goulotte de destination.

La rapidité et la fiabilité de l'exécution de cette phase est remarquable et l'étiquette "code à barres" est lue sans problème, quel que soit son sens, pourvu qu'elle soit dans un plan donné.



Mais ce n'est pas tout ! Dès qu'un instant de répit se présente, soit entre l'arrivée et le déchargement de deux camions, soit pendant une période où la chaîne de tri se vide, le système envoie automatiquement toutes les informations concernant les colis traités vers un ordinateur plus important. Celui-ci effectue les rapprochements nécessaires entre les informations qui viennent d'être saisies, et celles qui avaient été transmises par le client après la saisie initiale (minidisquette ou liaison téléphonique).

Lorsque l'opératrice termine la transaction, et donne son accord sur l'enregistrement du colis, une étiquette autocollante est imprimée. Celle-ci contient toutes les indications nécessaires à l'expédition, ainsi qu'un label imprimé en "code à barres", ce qui surprendrait nombre de professionnels de l'informatique s'ils savaient que l'imprimante utilisée est tout-à-fait banalisée.

L'étiquette n'est pas encore terminée que l'opératrice est déjà en train de saisir l'expédition suivante.

Etiquetés correctement, les colis prennent alors la route de Garonor où ils sont attendus afin d'être triés, puis repartent vers leur destination finale dans d'autres camions de transport.

La disquette créée lors de la saisie et contenant toutes les informations concernant le contenu de chaque camion, pourra soit être jointe à l'expédition et remise à l'arrivée, soit voir son contenu transmis à Garonor par liaison téléphonique.

A L'ARRIVEE ...

Maintenant, nous sommes à Garonor, le long d'un quai de déchargement, où se présente le camion contenant nos colis.

Sous ce grand hangar se trouve une gigantesque chaîne de tri de colis. Avant l'informatisation complète du système dans laquelle les petits TRS-80 ont en fait pris une part importante, un opérateur lisait l'adresse du destinataire sur le colis, en déduisait le département et, après avoir consulté un tableau et manoeuvré un clavier, provoquait le départ du colis vers une goulotte de destination correspondant à la région où devait être acheminé celui-ci.

La boucle est alors "bouclée" et il ne reste plus qu'à traiter ces informations : elles vont permettre un suivi très fin et en temps réel des expéditions ainsi que quelques exploitations comptables et statistiques.

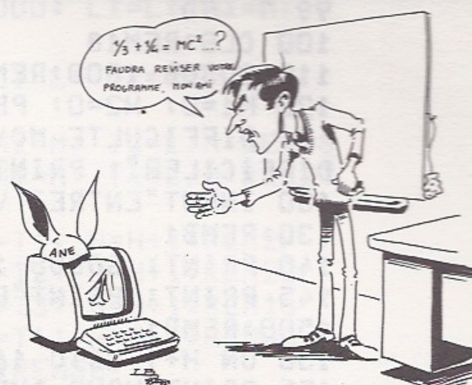
L'originalité de ce procédé consiste à avoir réussi à traiter, avec du matériel peu coûteux et donc accessible à toute entreprise si petite soit-elle, la quasi totalité des opérations d'expédition.

Un autre intérêt réside dans la modularité de l'application. Point n'est besoin pour commencer dans cette voie, de disposer de bascules électroniques, de lecteurs de code à barres ou de chaîne de tri de colis sophistiquées.

Il est toujours possible de peser le colis et d'entrer manuellement son poids sur le clavier de l'ordinateur. De même, à l'arrivée, on peut lire l'étiquette et organiser le tri en conséquence. Toutefois, ces perfectionnements techniques apportent, par une automatisation quasi totale de l'acheminement des colis, une très grande sécurité par l'élimination de toute erreur liée à une défaillance humaine, et par conséquent, un plus grand "confort" aux employés.

René Thomas

Un programme de classe



L'ordinateur peut être utilisé en matière d'enseignement de nombreuses façons, par exemple :

Par l'administration scolaire pour l'emploi du temps, la gestion des dossiers des élèves et des enseignants, la bureautique en général.

Par les professeurs pour la tenue des notes des élèves, la personnalisation des appréciations, la mise à jour des documents, la fabrication plus ou moins automatique d'exercices.

Par les élèves pour l'apprentissage des diverses matières.

Des ordinateurs sont utilisés avec succès pour enseigner des tranches complètes de cours, de manière interactive avec l'élève. On peut citer le système PLATO, ou le langage PILOT. Mais quand il s'agit d'exercices courts, point n'est besoin d'un langage particulier, le BASIC convient parfaitement.

Le but de cet article est de présenter deux exemples d'entraînement à des exercices mathématiques, entrant dans un même schéma de programme. Quand on écrit un programme pédagogique, on s'aperçoit assez vite que ce programme est autant consacré à la liaison ordinateur-élève qu'à l'exercice lui-même. On peut concevoir un tel programme en deux parties, l'une générale et aisément adaptable, et l'autre portant sur l'exercice lui-même.

Les programmes ci-dessous en sont une illustration. Mis au point par amusement, voici déjà pas mal de temps, ils étaient destinés à mes enfants.
Ils portent sur de calculs de fractions ainsi que sur la résolution d'équations linéaires à deux inconnues (à déterminant non nul).
Ils possèdent trois niveaux de difficultés, correspondant en fait à des données numériques plus ou moins grandes.

Ces programmes BASIC utilisent les facilités du TRS-80 :

- les catégories de variables sont déclarées au début, cela évite d'avoir à utiliser les signes spécifiques. Il suffit de se rappeler la partition effectuée au départ.
  - THEN est systématiquement omis après IF.
  - NEXT n'est pas suivi du nom de l'indice de la boucle.

Les programmes sont spécialisés à partir de la ligne 1000. La portion qui précède est commune, elle renvoie par GOSUB aux lignes propres de l'exercice :

GOSUB 1000 concerne l'affichage du titre de l'exercice.
GOSUB 1200 permet de fabriquer l'énoncé de l'exercice.
GOSUB 1400 affiche cet énoncé.
GOSUB 1500 détermine les bonnes réponses et compare à la solution de l'élève.
GOSUB 1600 affiche la bonne réponse.

Peu de chose à dire sur les exercices eux-mêmes.

PREMIER EXERCICE

Les fractions sont fournies simplifiées (on reconnaîtra l'algorithme d'Euclide dans le GOSUB 2000), mais l'ordinateur accepte de la part de l'élève toute fraction correcte non réduite. Signaler cette non réduction n'a pas été réalisé pour ne pas allonger le programme.

SECOND EXERCICE

Les équations sont obtenues à l'envers. On a ainsi des valeurs entières tant pour les coefficients que pour les racines. L'entrée aléatoire des seuls coefficients n'apportait rien, sinon quelques cas singuliers. De plus les formules de Cramer risquaient d'introduire des solutions à écriture fractionnaire compliquée ou décimale seulement approchée.

Jean Le Flour

1) - PARTIE COMMUNE
==================

```
10 ' EXERCICES MATHEMATIQUES  (J. LE FLOUR)  F6
15 '  F1
20 DEFSTR A-G: DEFINT H-N:REMC7
30 RANDOM:REM5C
50 A(0)="VOUS L'AVEZ FAIT EXPRES OU QUOI?":REME1
55 A(1)="C'EST TRES MAUVAIS, ATTENTION":REM61
60 A(2)="C'EST INSUFFISANT, VOUS DEVEZ TRAVAILLER":REM5A
65 A(3)="C'EST TROP MOYEN, IL FAUT VOUS ENTRAINER":REMD8
70 A(4)="C'EST HONNETE, VOUS POUVEZ VOUS AMELIORER":REM25
75 A(5)="C'EST BIEN, ESSAYER DE FAIRE ENCORE MIEUX":REME5
```

```
80 A(6)="C'EST TRES BIEN, CONTINUEZ DANS CETTE VOIE":REM64
85 A(7)="FELICITATIONS, VOUS ETES UN CHAMPION!":REM3E
99 '  9D
100 CLS:REM18
110 GOSUB 1000:REM40
120 H1=0: H2=0: PRINT"     1. EXERCICES FACILES": PRINT"
 2. DIFFICULTE MOYENNE": PRINT"     3. EXERCICES PLUS
DIFFICILES": PRINT:REM78
130 INPUT"ENTREZ VOTRE CHOIX";I: IF I<0 OR I>3 THEN
130:REMB1
140 PRINT: GOSUB 1200:REM34
145 PRINT: PRINT"ENTREZ VOS REPONSES": PRINT: GOSUB
1500:REMD1
150 ON H+1 GOTO 160,170,155:REMD4
155 PRINT"VOUS AVEZ TROUVE": GOTO 500:REMAB
160 PRINT"VOS REPONSES SONT  FAUSSES.": GOTO 400:REM1C
170 PRINT"VOS REPONSES SONT EN PARTIE EXACTES":REM50
400 PRINT: PRINT"    POUR RECOMMENCER L'EXERCICE, TAPEZ
<E>": PRINT"      POUR OBTENIR LA SOLUTION TAPEZ <S>":
PRINT:REM1D
410 G=INKEY$: IF G="" THEN 410:REMF8
420 IF G="E" GOSUB 1400: GOTO 145:REM70
430 IF G="S" GOSUB 1600: ELSE 410:REM19
500 PRINT: PRINT" SOUHAITEZ-VOUS": PRINT"           1.
CONTINUER AVEC UN EXERCICE DE MEME NIVEAU": PRINT"
2. CONTINUER AVEC UN EXERCICE DE DIFFICULTE DIFFERENTE":
PRINT"       3. INTERROMPRE LES EXERCICES":REM32
510 PRINT: INPUT"VOTRE CHOIX";I: PRINT:REMCD
520 ON I GOTO 140,530,540:REM61
530 GOSUB 600: PRINT: PRINT"     ET MAINTENANT": PRINT: GOTO
120:REMDB
540 GOSUB 600: FOR I=1 TO 4000: NEXT: CLS: PRINT@520,"A U
R E V O I R": PRINT: PRINT: PRINT: END:REM93
600 T=INT(200*H1/H2+.5)/10: PRINT"POUR CES EXERCICES AVEC
CE NIVEAU DE DIFFICULTE,": PRINT"             VOTRE NOTE ETAIT
"T"SUR 20": PRINT:REM93
610 PRINT A(INT(T/3+.5)): RETURN:REMDC
990 '  1F
995 '  + + + + + + + + + + +   1A
999 '  16
```

2) - PARTIE SPECIALISEE : CALCUL FRACTIONNAIRE
==

```
1000 PRINT"  ENTRAINEMENT AU CALCUL FRACTIONNAIRE": PRINT:
RETURN:REMCE
1199 '  4D
1200 PRINT: PRINT"     CALCULEZ LA SOMME, LA DIFFERENCE, LE
PRODUIT ET LE QUOTIENT DES FRACTIONS A ET B SUIVANTES
:":REM61
1210 N=7*I:REMD0
1220 L=RND(N)+5*(I-1): M=RND(N)+5*(I-1): GOSUB 2000: IF M=1
THEN 1220 ELSE LA=L: MA=M:REM4E
1230 L=RND(N)+5*(I-1): M=RND(N)+5*(I-1): GOSUB 2000: IF
L=LA OR M=MA OR M=1 THEN1230:REMA6
1240 LB=L: MB=M:REM8A
1399 '  84
```

```
1400 PRINT"A = "LA"/"MA"     B = "LB"/"MB: RETURN:REM62
1499 '  2D
1500 H=0: L=LA*MB+LB*MA: M=MA*MB: GOSUB 2000: LX=L: MX=M:
T1=L/M:REM02
1510 L=LA*MB-LB*MA: M=MA*MB: GOSUB 2000: LY=L: MY=M:
T2=L/M:REM7B
1520 L=LA*LB: M=MA*MB: GOSUB 2000: LZ=L: MZ=M: T3=L/M:REM28
1530 L=LA*MB: M=LB*MA: GOSUB 2000: LT=L: MT=M: T4=L/M:REM29
1535 '   FC
1540 PRINT" A + B   :";: GOSUB 1590: IF T=T1: H=H+1:REM2F
1550 PRINT" A - B   :";: GOSUB 1590: IF T=T2: H=H+1:REM22
1560 PRINT" A . B   :";: GOSUB 1590: IF T=T3: H=H+1:REM16
1570 PRINT" A / B   :";: GOSUB 1590: IF T=T4: H=H+1:REM0A
1580 H1=H1+H: H2=H2+4: H=INT(H/2): RETURN:REM58
1590 INPUT"  NUMERATEUR";L:PRINTCHR$(225)+CHR$(27);:INPUT"
  DENOMINATEUR";M: T=L/M: RETURN:REM30
1599 '   BB
1600 PRINT"A + B = "LX"/"MX"            A - B = "LY"/"MY:REM2E
1700 PRINT"A . B = "LZ"/"MZ"            A / B =
"LT"/"MT:RETURN:REMFF
1999 '  2A
2000 L2=ABS(L): M2=ABS(M): IF M2>L2: K=M2: M2=L2:
L2=K:REM04
2010 K=L2-M2*INT(L2/M2): IF K>0 L2=M2: M2=K: GOTO
2010:REM22
2020 L=L/M2: M=M/M2:RETURN:REM95
```

3) - PARTIE SPECIALISEE : CALCUL D'EQUATIONS

```
1000 PRINT"RESOLUTION DE SYSTEMES D'EQUATIONS LINEAIRES A 2
INCONNUES": PRINT: RETURN:REM68
1199 '  4D
1200 PRINT"VEUILLEZ RESOUDRE LE SYSTEME :": PRINT:REM9C
1210 I1=I:REMAA
1220 FOR I=1 TO 2: FOR J=1 TO 2:
K(I,J)=RND(4*I1+1)-5*(I1-1):REMCF
1225 IF K(I,J)=0: J=J-1: NEXT:REME1
1230 IF K(I,2)=K(I,1): J=J-1: NEXT: ELSE NEXT J,I:REM93
1240 IF K(1,1)*K(2,2)=K(1,2)*K(2,1) THEN 1220:REME7
1250 FOR I=1 TO 2: K(I,0)=RND(4*I1+1)-2*I1: IF K(I,0)=0:
I=I-1: NEXT: ELSE NEXT:REM18
1399 '  84
1400 FOR I=1 TO 2: PRINT K(I,1)"X";: IF K(I,2)<0: PRINT"
-"ABS(K(I,2));: ELSE PRINT" +"K(I,2);:REM00
1410 PRINT"Y ="K(I,1)*K(1,0)+K(I,2)*K(2,0): NEXT:
RETURN:REM1D
1499 '  2D
1500 INPUT"    VALEUR DE X ";K1: INPUT"    VALEUR DE Y
";K2: PRINT:REM8F
1510 H=0: IF K1=K(1,0): H=H+1:REME8
1520 IF K2=K(2,0): H=H+1:REM63
1530 H1=H1+H: H2=H2+2: RETURN:REM57
1599 '  BB
1600 PRINT"LES INCONNUES SONT  X ="K(1,0)" ET Y = "K(2,0):
PRINT: RETURN:REMD7
```

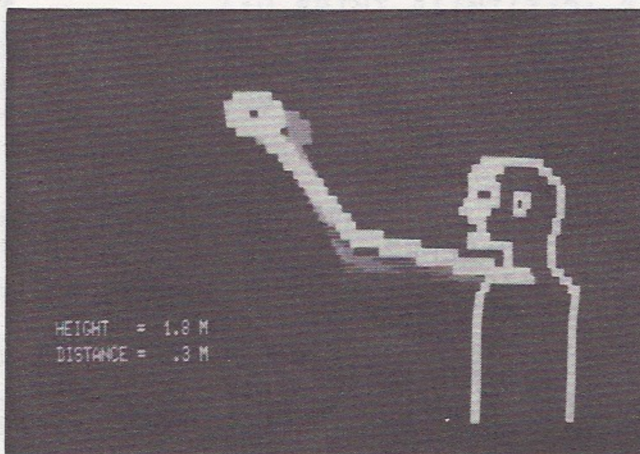


Pour les sportifs du clavier

Ce programme écrit par Timothy W. Smith, est une excellente démonstration des possibilités graphiques pouvant être atteintes sur le modèle 1 dont la résolution n'est pourtant pas très élevée. Bien que l'on puisse lui reprocher l'absence d'effets sonores (le coup de pistolet du départ, par exemple), ce jeu est très attractif et très amusant.

Le programme, comme l'indique son nom, fait participer le ou les joueurs (8 au maximum) aux dix épreuves classiques : 100 mètres, saut en longueur, lancement du poids, saut en hauteur, 400 mètres, 110 mètres haies, lancement du disque et du javelot, saut à la perche et 1500 mètres. Inutile de décrire l'état dans lequel se trouvent les participants à la fin des épreuves ... Il ne manque que le soigneur ! Quoi que l'on en pense, ce jeu est assez fatigant, tant sur le plan musculaire (les doigts surtout !) que sur le plan nerveux. En fait, c'est le clavier qui en souffre le plus ... et tout particulièrement les touches 1 et 2 !

Avant de participer au décathlon, chaque joueur a la possibilité de s'entraîner sur une ou plusieurs épreuves de son choix. Cela est nécessaire pour s'échauffer un peu ...
Par contre, lorsque les jeux sont ouverts, les points sont comptés et un classement général est affiché. Une médaille d'or récompensera le vainqueur !



Chaque épreuve est entièrement graphique et est accompagnée de quelques lignes de texte indiquant la marche à suivre et les touches utilisées : barre d'espace pour démarrer, flèche en bas pour planter la perche, flèche en haut pour lever le bras tenant le poids, etc ...

Selon les épreuves, certaines questions sont initialement posées : hauteur de la barre, hauteur de maintien de la perche, vitesse de rotation du marteau ...



Certaines épreuves sont particulièrement complexes. Le saut à la perche, par exemple, met en jeu cinq touches : flèches gauche et droite pour courir, flèche en bas pour planter la perche, flèche en haut pour donner le coup de rein et CLEAR pour lâcher la perche ... Personnellement, je n'ai jamais franchi la barre !

Parfois, le programme se montre très ironique, surtout s'il vous arrive quelques malheurs tels que tomber sur la tête, lancer le marteau dans les tribunes, ou lancer le poids à la verticale ! Il se montre également un juge très attentif : essayez, pour voir, de partir avant le coup de pistolet !

Le saut de haies est assez spectaculaire. En frappant - rapidement de préférence - sur les touches 1 et 2, on actionne les jambes du coureur. Lorsqu'une haie se présente, il faut laisser la touche enfoncée un bref instant pour que le coureur saute la haie. Si l'on a mal prévu son coup, le participant malchanceux heurte la haie qui s'incline et tombe, vous donnant de mauvais points ...

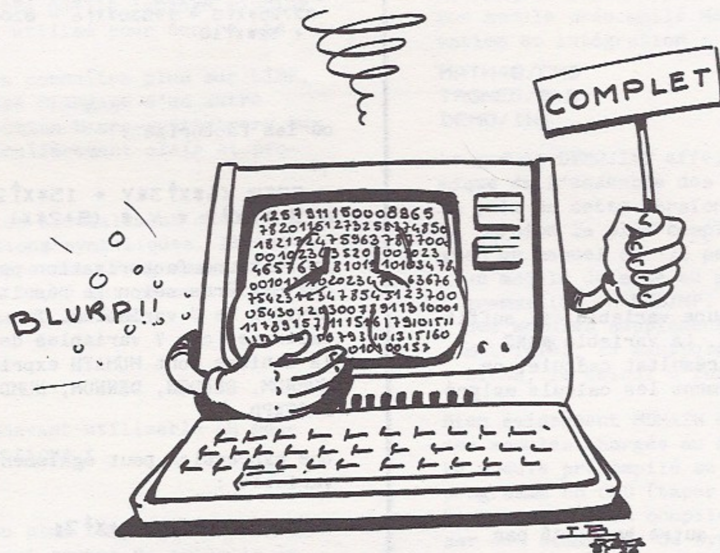
Les courses sur pistes peuvent admettre deux coureurs à la fois. Chaque joueur se déplace (en deux dimensions pour le 1500 mètres) selon des touches qui sont précisées au moment de l'épreuve. Si votre clavier est un tant soit peu fragile, c'est à ce moment-là qu'il rendra l'âme ... ou qu'il sera guéri de tous ses rebonds ! Le mien a passé l'épreuve du décathlon avec succès, mmoi ppass !!

Le programme est livré en version cassette ou disquette, et accompagné d'une notice explicative. La disquette qui résiste farouchement aux duplications "classiques", permet toutefois une seule copie. Chaque épreuve est lue à partir du disque alors qu'en version cassette, cinq sont chargées simultanément en mémoire.

En conclusion : un excellent programme pour les "sportifs du clavier" ...

Alain Pinaud

La boîte à outils mathématiques



Les ordinateurs ne savent pas calculer

Demandez à votre TRS-80 d'additionner ¼ et 1/3 et il vous répondra 0,583333 alors que le résultat est 7/12.
Essayez de calculer factorielle 30 (c'est-à-dire 30 x 29 x 28 x ... x 3 x 2 x 1) avec par exemple le petit programme BASIC suivant :

```
10 INPUT N
20 R=1
30 FOR I=1 TO N
40 R=R*I
50 NEXT I
60  PRINT N,R
70 GOTO 10
```

Vous obtiendrez pour N=30 : 2.65253E+32 au lieu des 33 chiffres du résultat exact :

265252859812191058636308480000000

Les interprêteurs tels que BASIC représentent toutes les informations sous forme de nombres compactés sur quelques octets, ce qui explique l'approximation obligatoire des résultats.
Mais il arrive souvent que l'on ait besoin de résultats rigoureusement exacts, et non approximatifs. Cela n'est possible qu'en utilisant un programme qui permet un calcul formel. Le programme MUMATH en est un excellent exemple, qui permet à l'utilisateur d'effectuer avec toute la rigueur nécessaire les calculs arithmétiques, algébriques, trigonométriques, différentiel et intégral, comme on les effecturait (beaucoup plus laborieusement !) à la main.

Un exemple d'utilisation

Le signe d'invite de MUMATH est "?" qui indique qu'il est prêt à accepter une commande. Vous tapez alors une commande ou une opération suivie de ";".

Le programme exécute les calculs puis imprime sur la ligne suivante "@" suivi du résultat :

```
? (1/3+9/27)/3;

@ 2 / 9
```

(vérifiez, c'est bien 2/9 le résultat et non sa valeur approchée 0,222222).
On peut aussi facilement élever un nombre à une puissance, et obtenir tous les chiffres significatifs :

```
? 3↑50;

@ 717897987691852588770249
```

(et non 7,17898E+23).
Les calculs de factorielle sont aussi dans les compétences du programme, même s'ils prennent plus de temps :

```
? 100!;

@ 9332621544394415268169923885626670049071596826438162146859296
3895217599993229915608941463976156518286253697920827223758251185
21091686400000000000000000000000
```

(5 secondes de calcul)

```
? 200!;

@ 788657867364790503552363213932185062295135977687173263294742533
24435944996340334292030428401198462390417721213891963883025764
27902426371050619266249528299311134628572707633172373969889439
22445621451664240254033291864131227428294853277524242407573903240
32125740557956866022603190417032406235170085879617892222278962
370389737472000000000000000000000000000000000000000000000
```

(20 secondes de calcul)

Une seule condition : la longueur des nombres ne doit pas dépasser 611 chiffres ce qui représente une précision virtuellement infinie ! Ainsi le programme refuse-t-il de calculer factorielle 300, qui comporte trop de chiffres :

```
? 300!;

@ FALSE
```

L'utilisation de variables permet une grande souplesse de calcul. Le signe d'affectation est ":" et non "=" comme en BASIC :

```
?
  N: 30 ;
@ 30

?
  M: 12 ;
@ 12

?
  N! / ((N-M)!*M!)
@ 86493225
```

Pour visualiser le contenu d'une variable, il suffit de taper son nom suivi de ";". La variable #ANS contient toujours le dernier résultat calculé, ce qui permet d'enchaîner facilement les calculs malgré la longueur des résultats.

```
? 25**#ANS;

@ 2162330625
```

Voulez-vous calculer dans une autre base (16 par exemple) ? rien de plus simple :

```
?
  RADIX (16)
@ A
```

(A représente l'ancienne base dans la nouvelle, c'est à-dire 10 en base 16). Le contenu de toutes les varaibles est automatiquement converti. Vous disposez alors d'une calculatrice hexadécimale :

```
?
  7CB0 - 2*12EF
@ 56A2
```

(pour revenir en base 10, taper RADIX(0A);)
MUMATH peut également calculer les Plus Grands Communs Diviseurs :

```
? GCD(567,453);

@ 3
```

et les Plus Petits Communs Multiples

```
? LCM(139,675);

@ 93825
```

La trigonométrie n'a pas de secrets pour MUMATH, qui connaît également les constantes telles que #PI (3,14159...) :

```
?
  SIN (37**#PI/3) ;
@ 3↑(1/2) / 2
```

(remarquez que là encore, on obtient un résultat exact, et non une valeur numérique approchée) ainsi que #E (base des logarithmes Népériens) et #I (pour les nombres imaginaires)

```
? #E↑(3 * #I * #PI/2);

@ - #I
```

Voulez-vous faire de l'algèbre ? MUMATH accepte les variables indéfinies dans les expressions. BASIC signalerait une erreur alors que MUMATH les considère comme des inconnues et simplifie de telles expressions :

```
? 6*X↑5/(9*X);

@ 2*X↑4 / 3
```

MUMATH développe très facilement des expressions algébriques :

```
?
  (2*X↑2 - 5*X + 4)↑5
@ 1024 - 6400*X + 18560*X↑2 - 32800*X↑3 + 39060*X↑4 -
 32725*X↑5 + 19530*X↑6 - 8200*X↑7 + 2320*X↑8 - 400*X↑
9 + 32*X↑10
```

ou les factorise :

```
?
  FCTR (6*X↑3*Y + 15*X↑2*Y)
@ 3 * X↑2 * Y * (5+2*X)
```

En fait une factorisation peut s'effectuer de bien des manières selon le résultat souhaité, grâce au contenu de 7 variables. Ils vous faudra en fait maîtriser ces 7 variables de contrôle pour définir la manière dont MUMATH exprimera les résultats : NUMNUM, DENDEN, DENNUM, NUMDEN, BASEXP, EXPBAS, PWREXPD.

Une expression peut également être affectée à une variable :

```
? TOTO:P*SIN(X)+X↑2;

@ P*SIN(X) + X↑2
```

Le calcul différentiel et la calcul intégral sont les possibilités les plus étonnantes de MUMATH. Dérivons TOTO par rapport à X :

```
? DIF(TOTO,X);

@ 2*X + P*COS(X)
```

MUMATH est également très brillant en calcul intégral. Intègrons par exemple $\frac{1}{ax2+bx+c}$ par rapport à x :

```
? INT(1/(A*X↑2+B*X+C),X);
```

Après dix secondes de calcul, il demande, puisque dans le cas particulier de la fonction intégrée ceci est nécessaire :

```
@ IS SIGN (4*A*C-B↑2)    -1 $ 0 $ OR 1 $ ?
```

Si vous répondez +1, pour indiquer que le signe du discriminant est positif, vous obtenez trois secondes plus tard le résultat :

```
 2*ATAN(2*A*X/(4*A*C-B↑2)↑(1/2)+B/(4*A*C-B↑2)↑(1/2))
/ (4*A*C-B↑2)[(1/2)
```

que l'on factorise :

```
? FCTR(#ANS);

@ 2*ATAN((B+2*A*X)/(4*A*C-B↑2)↑(1/2)) / (4*A*C-B↑2)↑(
1/2)
```

Ces quelques exemples montrent la simplicité d'utilisation de MUMATH, qui bien au delà de son caractère spectaculaire, peut constituer un outil précieux qui se chargera de tous les calculs laborieux pour lesquels un résultat exact doit être obtenu.
Cet outil est notamment très attrayant pour l'enseignement (que les professeurs soient rassurés ! MUMATH donne les résultats mais pas les solutions).

Langage d'un autre type : MUSIMP

MUMATH est en fait fourni sous forme de programmes sources écrits en MUSIMP. L'utilisateur dispose donc en prime de l'interprêteur MUSIMP, ce qui est particulièrement intéressant car cela lui permet de modifier les fonctions existantes, ou d'ajouter ses propres fonctions. Le langage MUSIMP est une variante du langage LISP, bien connu pour ses applications en Intelligence Artificielle. LISP peut être considéré comme le "langage machine" de l'intelligence artificielle, utilisé pour créer des langages plus évolués, de la même manière que le langage machine d'un microprocesseur est utilisé pour écrire les interpréteurs BASIC.
(Pour ceux qui veulent en connaître plus sur LISP, citons notamment l'ouvrage "Langage d'un autre type : LIST" de la collection Micro-ordinateurs aux éditions Eyrolles, particulièrement clair et progressif).

Le langage MUSIMP est plus spécialement destiné à la manipulation des expressions symboliques. Il est relativement facile à apprendre. Programmons par exemple, la fonction BIN (N,M) qui permettra de calculer les coefficients du bicône.

```
? FUNCTION BIN (N,M),
   N! /((N-M)!*M!),
 ENDFUN $
```

La fonction BIN est dorénavant utilisable au même titre que les autres fonctions :

```
? BIN (30,12);
@ 86493225
```

Prenons un exemple un peu plus compliqué : programmons la fonction TAYLOR qui permet de calculer le développement en série de TAYLOR d'une fonction (EXPN) par rapport à une variable (X), autour d'un point (A), à un certain ordre (N) :

```
FUNCTION TAYLOR (EXPN, X, A, N,
      % Local % J, C, ANS, NUMNUM, DENNUM),
  NUMNUM: DENNUM: 30,
  J: ANS: 0,
  C: 1,
  LOOP
    ANS: ANS + C * EVSUB (EXPN, X, A),
    WHEN J=N, ANS EXIT,
    EXPN: DIF (EXPN, X),
    J: J + 1,
     C: C * (X-A) / J,
  ENDLOOP,
ENDFUN $
```

Utilisons cette nouvelle fonction par exemple pour déterminer le développement à l'ordre 6 de SIN (X) autour de zéro :

```
?
 TAYLOR (SIN(X), X, 0, 6) ;
@ X*COS(0) - X^2*SIN(0)/2 - X^3*COS(0)/6 + X^4*SIN(0)
/24 + X^5*COS(0)/120 - X^6*SIN(0)/720 + SIN(0)
```

(10 secondes de calcul)
La puissance du langage LISP et donc de MUSIMP, associée à MUMATH, permet de créer virtuellement n'importe quelle fonction.
Malheureusement MUSIMP n'incorpore pas d'éditeur, sans doute pour gagner de la place en mémoire lorsque le programme est chargé : ainsi, pour modifier le texte d'une fonction, il faut retaper entièrement sa définition. Sauf si l'on utilise l'éditeur externe, au prix de quelques manipulations.

Les versions de MUMATH

Il existe pour le TRS-80 deux versions de MUMATH : la version simplifiée et la version complète.
La version simplifiée que nous avons testée, est livrée avec un manuel abrégé de 75 pages (en anglais) et deux minidisquettes (une disquette pour les systèmes 32K, une autre pour les 48K).

La disquette pour les systèmes 32K comporte :
. un module précompilé MATH32/CMD qui comprend l'interprêteur MUSIMP, les fonctions arithmétiques et algébriques ainsi que la résolution d'équation.
. un module de logarithmes (LOG/ALG)
. un module de trigonométrie positive (TRGPOS/ALG).
. un module de trigonométrie négative (TRGNEG/ALG).

La disquette pour système 48K comporte en plus dans son module précompilé MATH48/CMD les fonctions dérivation et intégration :

```
MATH48/CMD          LOG/ALG
TRGNEG/ALG          TRGPOS/ALG
DEMO/INT
```

Le module DEMO/INT effectue une démonstration dynamique de l'ensemble des possibilités de MUMATH.
Le prix de cette version est de 690 F.
La version la plus complète de MUMATH est fournie avec un manuel de 175 pages. Elle comporte de plus :
. un module de mise au point (TRACE) pour la programmation en MUSIMP.
. des modules supplémentaires permettant le calcul matriciel, la manipulation et la résolution d'équations, le calcul d'intégrales définies.

Bien évidemment MUMATH ne dispose des ressources que des modules chargés au moment de l'utilisation.
Le module précompilé se charge simplement comme tout programme en CMD (taper MATH32 ou MATH48).
Les modules (non compilés) peuvent alors se charger par des commandes du type :

```
? RDS(LOG,ALG);
? RDS(TRGPOS,ALG);
? RDS(TRGNEG,ALG);
```

Le module de démonstration, qui nécessite au préalable le chargement des trois modules précédents, se charge de la même manière :

```
? RDS(DEMO,INT);
```

Ces opérations de chargement sont relativement longues. Le chargement du module LOG/ALG nécessite cinquante secondes environ, et celui des modules TRGPOS/ALG et TRGNEG/ALG nécessite deux minutes pour chacun.
Ce qui demande en tout environ cinq minutes. (non non, ce n'est pas une cassette, c'est bien un disque !).
D'autre part les temps de calcul de MUMATH varient à l'inverse de l'espace mémoire inutilisé : les modules ne devront être chargés que si nécessaire.

Conclusion

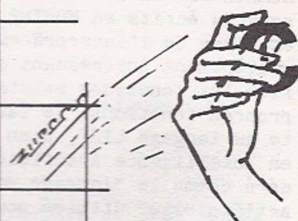
L'ensemble MUMATH / MUSIMP est un outil très impressionnant qui comble une lacune dans le spectre des applications possibles des ordinateurs en général ; sa disponibilité sur un OI bon marché devrait intéresser de nombreux utilisateurs professionnels, notamment ceux qui ont de nombreux calculs scientifiques.

La documentation est bonne, mais on ne se sent à l'aise qu'après un minimum de pratique. Le langage MUSIMP n'est que sommairement décrit mais un manuel plus complet semble disponible sur commande : je pense qu'en fait, c'est surtout MUMATH qui intéressera les utilisateurs, et que MUSIMP n'en concerne qu'une partie plus réduite.

A montrer absolument à tous les sceptiques de l'informatique individuelle qui ne jurent que par les gros systèmes !

Richard SCHOMBERG

Des nombres sur la grille, MATHI tel est



La partie se poursuit jusqu'à ce que le joueur ou l'ordinateur ne puisse plus jouer et le gagnant est celui qui a obtenu le maximum de points.

Ce jeu comporte quatre niveaux de difficulté :

- Niveau 1 : L'ordinateur "réfléchit" peu.
- Niveau 2 : Identique au précédent mais il y a "création" aléatoire de nombres dans la grille, aux places laissées libres pendant le cours du jeu.
- Niveau 3 : L'ordinateur "réfléchit" davantage.
- Niveau 4 : Combinaison des niveaux 2 et 3. De plus, certains nombres "créés" sont inconnus (X ou -X).

Attention : les signes "-" ne doivent pas être frappés. Pour "prendre" -5 ou -X par exemple, frapper : 5 ou X.

Le jeu que nous vous proposons ici, vous permettra d'exercer vos facultés intellectuelles en jouant contre l'ordinateur. Attention : c'est un concurrent redoutable et qui ne fait preuve d'aucune pitié !

Une grille carrée (8 x 8) comporte 64 nombres, dont l'un est remplacé par des étoiles. Ces nombres sont toujours les mêmes au départ, mais leur répartition dans la grille est aléatoire.

Le joueur choisit un nombre sur la LIGNE où se trouvent les étoiles et celles-ci (que l'on ne voit plus à ce moment) remplacent alors le nombre choisi. Les nombres choisis sont cumulés pour former le total du joueur.

L'ordinateur joue de la même façon, mais choisit un nombre dans la COLONNE où se trouvent les étoiles, déterminant alors la nouvelle position de celles-ci.

Roger Sawicki

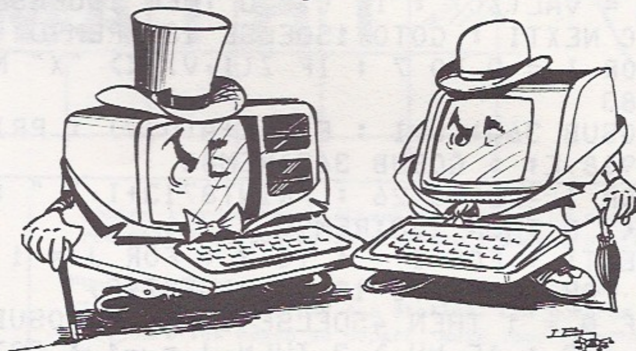
```
10 RANDOM : CLS : DEFINTA-Y : DEFSTRZ : DIMA(7,7),Z(7,7),B(7) : PRINT@
275,"***  M A T H I C  ***":REM08
20 PRINT@708,"ATTENTION !  LES SIGNES  -  NE DOIVENT PAS ETRE TAPES":P
RINT@528,"NIVEAU   1 , 2 , 3 , 4      "; : INPUT ZV:REMFE
30 NV = VAL(ZV) : IF NV < 1 OR NV > 4 RUN ELSE CLS : FOR I = 22 TO 104
 : SET(I,2) : SET(I,27) : NEXTI:REM65
40 FOR I = 3 TO 26 : SET(22,I) : SET(104,I) : NEXTI : X = RND(7) : Y =
 RND(7) : PRINT@55,"NIVEAU"; : PRINT@121,NV;:REM69
50 DATA 2,-3,8,-1,12,4,-9,10,-3,6,-7,-11,14,-1,2,4,-5,2,-3,8,4,18,10,-
1,6,-7,4,10,-9,-5,-11,2,4,14,2,-3,8,10,-1,6,-7,4,-9,6,-1,2,-5,8,-13,12
,-1,2,-3,6,4,-5,-1,16,6,12,-5,8,-3,-7:REMD8
60 FOR I = 0 TO 7 : K = I+X-8*INT((I+X)/8) : FOR J = 0 TO 7 : L = J+Y-
8*INT((J+Y)/8) : READ A(K,L) : PRINT@78+5*I+64*J,"."; : NEXTJ,I : FOR
I = 0 TO 7 : FOR J = 0 TO 7:REM71
70 VA=A(I,J) : PRINT@77+5*I+64*J-INT(ABS(VA/10)),VA; : NEXTJ,I : U = R
ND(8)-1 : V = RND(8)-1 : PRINT@76+5*U+64*V," *** "; : A(U,V) = 0:REM11

80 PRINT@66,EE+FF; : PRINT@0,"PARTIE"; : IF EE+FF > 1 PRINT@6,"S";:REM
65
90 PRINT@194,EE; : PRINT@128,"PERDUE"; : IF EE > 1 PRINT@134,"S";:REMO
E
100 PRINT@322,FF; : PRINT@256,"GAGNEE"; : IF FF > 1 PRINT@262,"S";:REM
21
110 PRINT@905,"VOS POINTS  :                         ORDINATEUR  :"; : PRINT@713,"
VOULEZ-VOUS COMMENCER (1) ";:REM25
120 INPUT ZK : FOR I = 0 TO 29 : PRINT@713+I," "; : IF ZK <> "1" PRINT
@76+5*U+64*V,"  .  "; : GOTO 190:REMF9
130 FOR I = 0 TO 7 : IF A(I,V) <> 0 THEN 150ELSE NEXTI : PRINT@713,"VO
US NE POUVEZ PAS JOUER                  ";:REM18
```

```
140 FOR I = 1 TO 150 : NEXTI : IF Q = 1 THEN 450ELSE Q = 1 : GOSUB 360
: GOTO 190:REMD9
150 PRINT@76+5*U+64*V," *** "; : Q = 0 : PRINT@713,"VOUS JOUEZ HORIZON
TALEMENT        " : INPUT ZC : IF ZC = "X" THEN 170:REMA3
160 C = VAL(ZC) : IF C = 0 THEN 150ELSE FOR I = 0 TO 7 : IF ABS(A(I,V)
) <> C NEXTI : GOTO 150ELSE 180:REM90
170 FOR I = 0 TO 7 : IF Z(I,V) <> "X" NEXTI : GOTO 150ELSE Z(I,V) = "
":REM83
180 GOSUB 360: U=I : E = E+A(U,V) : PRINT@851,A(U,V); : A(U,V) = 0 : P
RINT@918,E; : GOSUB 360:REM86
190 FOR I = 0 TO 26 : PRINT@713+I," " : NEXTI : FOR I = 0 TO 7 : IF A(
U,I) <> 0 THEN 220:REM9D
200 NEXTI : FOR I = 1 TO 3 : FOR J = 1 TO 19 : PRINT@720,"PASSE"; : NE
XTJ : FOR J = 1 TO 10 : PRINT@720,"     "; : NEXTJ,I:REM2C
210 IF Q = 1 THEN 450ELSE Q = 1 : GOSUB 360: GOTO 130:REM00
220 Q = 0 : IF NV > 2 THEN J = -1 : GOTO 280:REM12
230 N = -20 : FOR I = 0 TO 25 : PRINT@713+I," "; : NEXTI:REM08
240 FOR I = 0 TO 7 : IF A(U,I) <> 0 AND A(U,I) > N THEN T = I : N = A(
U,I):REM09
250 NEXTI:REM36
260 F = F+N : PRINT@944,F; : P = 76+5*U+64*T : FOR I = 1 TO 3 : FOR J
= 0 TO 10 : PRINT@P+1,N; : NEXTJ:REM5D
270 FOR J = 0 TO 10 : PRINT@P," *** "; : NEXTJ,I : V = T : A(U,V) = 0
: Z(U,V) = " " : GOSUB 360: GOTO 130:REME9
280 J = J+1 : IF J > 7 THEN 330:REM02
290 IF A(U,J) = 0 B(J) = -99 : GOTO 280:REM99
300 B(J) = A(U,J) : N = -50 : FOR I = 0 TO 7 : IF A(I,J) = 0 OR I = U
NEXTI : IF A(U,J) > 0 T = J : GOTO 350ELSE B(J) = -50:REM52
310 FOR I = 0 TO 7 : IF I <> U AND A(I,J) > N THEN N = A(I,J):REM72
320 NEXTI : B(J) = B(J)-N :GOTO 280:REMA9
330 N = -99 : FOR I = 0 TO 7 : IF B(I) > N THEN N = B(I) : T=I:REM19
340 NEXTI:REMDB
350 FOR I = 0 TO 7 : B(I) = 0 : NEXTI : N = A(U,T) : GOTO 260:REM96
360 PRINT@76+5*U+64*V,"  .  "; : IF NV = 1 OR NV = 3 RETURN:REM59
370 B = RND(10) : D = RND(8)-1 : G = RND(8)-1 : IF B/NV = INT(B/NV) OR
 A(D,G) <> 0 OR D = U OR G = V RETURN:REM30
380 S = RND(10)*2+1 : FOR R = 0 TO 7 : IF ABS(A(R,G)) = S OR ABS(A(D,R
)) = S THEN 380:REM76
390 NEXTR : A(D,G) = S : OO = 77+5*D+64*G-INT((2+S)/12) : XX = RND(3)
: IF NV <> 2 AND XX <> 1 THEN 410ELSE FOR SS = 1 TO 4:REM98
400 FOR H = 1 TO 5 : PRINT@OO,"    "; : NEXTH : FOR H = 1 TO 5 : PRINT
@OO,S; : NEXTH,SS : RETURN:REM0A
410 FOR II = 0 TO 7 : IF Z(II,G) = "X" THEN 380:REM0A
420 NEXTII : OO = 78+5*D+64*G : FOR S = 1 TO 4 : FOR H = 1 TO 5 : PRIN
T@OO," "; : NEXTH : FOR H = 1 TO 5 : PRINT@OO,"X";:REM7E
430 NEXTH,S : YY = RND(2)*2-3 : A(D,G) = A(D,G)*YY : Z(D,G) = "X" : IF
 YY = -1 PRINT@OO-1,"-";:REMFA
440 RETURN:REMB5
450 PRINT@705,"          VOUS AVEZ "; : IF E < F PRINT@725,"PERDU !
   "; : EE = EE+1 : GOTO 470:REMF8
460 PRINT@725,"GAGNE !    "; : FF = FF + 1:REM50
470 FOR I = 0 TO 7: FOR J = 0 TO 7 : Z(I,J) = " " : NEXTJ,I : PRINT@7
33,"     ";:REMB7
480 PRINT@768,"POUR UNE AUTRE PARTIE , QUEL NIVEAU  ";:REM3E
490 INPUT ZV : E = 0 : F = 0 : RESTORE : GOTO 30:REMDF
```



Modèle 1, modèle 3, quelles différences ?



1 - LE MATERIEL

La première différence entre le modèle 1 et le modèle 3 se remarque immédiatement en regardant les deux machines : la première est un système modulaire, les divers modules étant reliés par des câbles, peu esthétique et souvent à la source de mauvais contacts, alors que la deuxième a une présentation intégrée plus fiable, plus pratique si l'ordinateur doit être transporté fréquemment, et prenant moins de place. En outre le raccordement au secteur d'un modèle 3 nécessite beaucoup moins de prises de courant qu'un modèle 1, l'unité centrale, l'écran vidéo et les deux premiers lecteurs de disquettes étant reliés ensemble.
Aucun changement par contre dans les diverses capacités de mémoire centrale proposées: la version minimale est 4K, puis on trouve des versions de 16K, 32K ou 48K. Le passage du modèle 3 à 32K ou 48K ne nécessite plus d'interface d'extension, mais les supports des circuits de mémoire additionnels sont directement prévus sur le circuit imprimé.

Le micro-processeur est toujours un Z80, mais sa vitesse de travail est 2 MHz au lieu de 1,77 MHz, ceci pouvant créer des problèmes pour les programmes travaillant en temps réel avec des boucles de temporisation extrêmement précises.

L'écran vidéo est de la même taille que celui du modèle 1, soit 12 pouces (30 cm), mais n'est actuellement fourni qu'en noir et blanc.

Le clavier est parfaitement identique sur les deux modèles : 53 touches sur le clavier principal plus un clavier numérique de 12 touches. Une amélioration notable : la touche RESET n'est plus cachée à l'arrière de la machine, mais bien visible sur le dessus du clavier (peut être un peu trop ! car celà n'incite pas à la prudence).

Le magnétophone à cassettes n'est plus fourni en standard comme c'était le cas du modèle 1, mais il peut être acheté en option.

L'interface parallèle pour imprimante est fournie, même en version de base, la connexion étant faite par un câble simple, sans interface. Ceci permet aux possesseurs de 16K de connecter une imprimante, sans pour cela augmenter le prix. Par contre l'interface série RS232C reste en option.

Le modèle 3 peut être équipé de lecteurs de mini-disquettes 13 cm au nombre maximum de 4, les deux premiers ayant leur place réservée dans la machine à droite de l'écran, les deux suivants étant placés à l'extérieur et reliés à l'unité de contrôle par un câble.

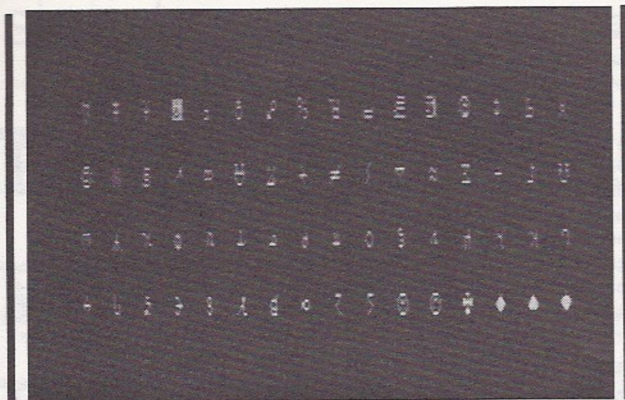
Le modèle 3 est équipé en standard d'une horloge interne, alors qu'elle n'était que dans l'interface d'extension du modèle 2.

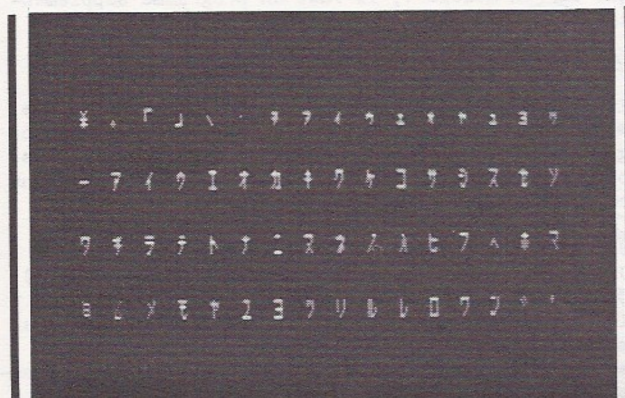
2 - LES CARACTERISTIQUES

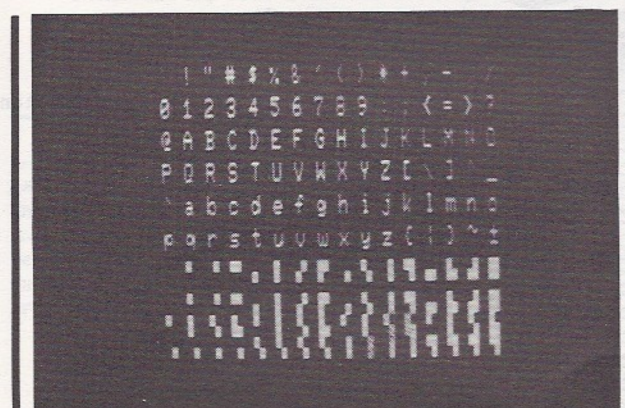
L'affichage peut se faire en deux modes : 16 lignes de 64 caractères ou 16 lignes de 32 caractères sélectables par logiciel (PRINT CHR$ (23)) comme sur le modèle 1. Par contre, le jeu de caractères fourni a été sérieusement remanié : on dispose maintenant des majuscules et des minuscules à l'écran en standard, alors que pour obtenir les minuscules avec un modèle 1, il faut ajouter un kit à l'intérieur de la machine. Les 32 premiers caractères, dont les numéros ASCII vont de 0 à 31 sont des codes de contrôle lorsqu'ils sont utilisés dans une instruction PRINT, mais deviennent des caractères lorsqu'ils sont placés dans la RAM écran par une instruction POKE (voir photo n° 1).

Les caractères dont le numéro ASCII vont de 192 à 255 représentent normalement les codes de compression d'espaces, comme sur le modèle 1, mais deux instructions, PRINT CHR$ (21) et PRINT CHR$ (22), permettent de passer soit à un jeu de caractères semi-graphiques (voir photo n° 2), soit au jeu des caractères japonais KANA (voir photo n° 3). Le reste des caractères est le jeu ASCII standard sauf le n° 127 qui est ± (voir photo n° 4).

Le caractère représentant le curseur peut être choisi par programme et peut être fixé ou clignotant.







Par exemple POKE 16412,1 permet d'avoir le curseur fixe alors que POKE 16413,0 permet d'avoir un curseur clignotant, pour changer le curseur : POKE 16419,X où X est le numéro ASCII du caractère choisi.

Le modèle 3 offre aussi la possibilité de protéger jusqu'à 7 lignes contre le "scrolling". Pour protéger 4 lignes il suffit de mettre l'instruction POKE 16916,4.

L'interface cassette a été modifiée pour permettre la lecture à deux vitesses : 500 et 1500 bauds. La vitesse lente permet à un modèle 3 de lire des cassettes enregistrées pour modèle 1. La sélection entre les deux vitesses peut se faire soit à l'allumage de la machine (CASS ?) soit plus tard en utilisant un POKE 16913,0 ou 1. Les codes erreurs donnés lors du chargement d'une cassette sont maintenant : C* ,'chechsum error' qui sanctionne une erreur de chargement d'une cassette system, D*, 'Data error' qui notifie une erreur dans un programme Basic, BK, qui apparaît lorsque l'utilisateur a appuyé sur la touche BREAK pendant un chargement. Le clignotement de la seconde étoile se fait maintenant tous les 64 caractères, alors que sur un modèle 1, il se fait tous les 256 caractères en "system" et à chaque fin de ligne en Basic. Il semble qu'il n'est pas possible de brancher un second magnétophone à cassettes comme sur un modèle 1.

L'interface imprimante a reçu aussi quelques améliorations : il est possible de spécifier la longueur maximum d'une ligne, de faire une recopie de l'écran en appuyant sur les 3 touches SHIFT↓ et * simultanément, d'arrêter cette recopie en appuyant sur la touche BREAK (on peut aussi profiter de cette possibilité dans un programme Basic ou langage machine).

Les sous-programmes élémentaires de gestion de l'interface série RS232C sont en mémoire morte, et paramètrables, alors que sur le modèle 1, ils sont chargés en mémoire vive depuis une cassette avec l'interface.

Le modèle 3 permet en principe (en fait, cela ne marche pas du tout en version disque) de dériver la sortie de caractères d'un périphérique vers un autre. Les périphériques pouvant être déroutés sont le clavier, KI, l'écran, DO, l'imprimante, PR, la RS232C en émission, RO et la RS232C en réception, RZ. Par exemple pour envoyer les caractères destinés à l'écran sur l'imprimante :

```
POKE 16930, ASC("D") : POKE 16931, ASC("O")
POKE 16928, ASC("P") : POKE 16929, ASC("R")
POKE 16526, 108 : POKE 16527,0 : X=USR(0).
```

De l'étude qui précède, on peut déduire que le modèle 3 représente une bonne amélioration du modèle 1, en offrant aux utilisateurs de 16K cassettes des possibilités qui jusque là étaient réservées aux possesseurs de modèle 1 équipé de lecteurs de disquettes.

3 - LA COMPATIBILITE EN BASIC

Le Basic du modèle 3 est exactement semblable à celui du modèle 1. Ceci permet de supposer que tous les programmes écrits en Basic pour le modèle 1 peuvent tourner sans aucune modification sur le modèle 3 (à condition toutefois qu'il n'y ait pas dans le programme de PEEK ou de POKE "bizarres" !).
De même tous les programmes Basic écrits pour le modèle 3, mais ne profitant pas de nouvelles possibilités (enregistrement à 1500 bauds, jeux de caractères nouveaux, ...) pourront tourner sur un modèle 1. La seule exception envisageable sont les programmes Basic qui utilisent les deux magnétophones autorisés par un modèle 1 avec interface d'extension.

Le seul ajout que l'on peut constater au Basic du modèle 3 est dû au fait qu'il possède en standard une horloge interne. La variable TIME$ (connue des possesseurs de modèle 1 disk) est maintenant utilisable, mais la remisé à l'heure ou à la date se fait par des POKEs (car l'horloge est initialisée à 00/00/00 et 00:00:00 à l'allumage).

4 - LA COMPATIBILITE EN LANGAGE MACHINE

Puisque le modèle 3 semble parfaitement compatible avec le modèle 1 en Basic, on peut se demander s'il en est de même pour les programmes écrits en langage machine. Pour assurer cette compatibilité il faut que les principales adresses des sous-programmes en

ROM soient inchangées. Les adresses principales sont celles permettant la gestion de l'écran, du clavier, de l'imprimante, du magnétophone.

La gestion de l'écran

La mémoire réservée à l'écran vidéo est située de 3C00H à 3FFFH sur les deux modèles. Pour afficher un caractère à l'écran, il y a plusieurs formules identiques sur les deux modèles : un LD direct, un CALL 0033H, avec le numéro ASCII du caractère dans le registre A, celui-ci étant affiché à la position indiquée par les cases mémoires 4020H-4021H, ou un CALL 033AH qui outre l'affichage du caractère détermine aussi la position du curseur dans la ligne et le stocke dans 40A6H.
Pour effacer l'écran on peut faire appel au CALL 01C9H. Toutefois seul le modèle 3 peut afficher un message entier par un CALL 021BH.

La gestion du magnétophone

Si la gestion de l'écran et du clavier semblent à peu près semblable, sur les deux modèles, il n'en va pas tout à fait de même pour la gestion du magnétophone. Il existe quelques différences et il se pourrait que quelques cassettes bien protégées du modèle 1 refusent de charger sur un modèle 3.

Pour mettre en route le magnétophone, sur un modèle 1, on peut utiliser plusieurs techniques. La plus simple étant de placer dans le registre A le n° du magnétophone (0 ou 1) et CALL Ø212H, ou encore pour le magnétophone n° 1, LD HL,Ø CALL Ø1FEH. Pour arrêter le magnétophone : CALL Ø1F8H.

Sur le modèle 3, tous ces sous-programmes sont inutiles, puisqu'il n'y a plus qu'un seul magnétophone. Par souci de compatibilité, on peut toujours faire appel au CALL Ø212H, mais cela ne fait pas grand chose (XOR A RET), l'arrêt du magnétophone étant maintenu à CALL Ø1F8H.

Pour la lecture des cassettes, nous avons besoin de deux sous-programmes l'un lisant le "leader" (256 zéros + A5H), CALL Ø296H, l'autre permettant la lecture d'un octet, CALL Ø235H.

Pour l'écriture des cassettes, nous avons aussi besoin de deux sous-programmes : l'un écrivant le "leader", CALL Ø284H, l'autre écrivant un octet, CALL Ø264H.

Sur les deux points, les deux modèles sont en accord, par contre certains sous-programmes tels que CALL Ø241H, lire un bit, ou CALL Ø1D9H, écrire un bit, sont déplacés sur le modèle 3. Le sous-programme permettant de faire clignoter une étoile en haut et à droite de l'écran qui dans le modèle 1 s'appelle par CALL Ø22CH a été déplacé, à la place on trouve un XOR A ,RET.

La gestion du clavier

La mémoire réservée au clavier est située de 3800H à 3BFFH sur les deux modèles, la matrice de codage étant la même. Pour acquérir un ou plusieurs caractères du clavier le modèle 1 et le modèle 3 ont les mêmes sous-programmes : le premier, qui est équivalent au INKEY du Basic est CALL 002BH. Le second, CALL 0049H, explore le clavier et attend que l'utilisateur ait appuyé sur une touche. Le troisième, CALL 0040H ou CALL 05D9H, est équivalent au INPUT du Basic, il permet l'entrée d'une chaine de caractères.

Le modèle 3 possède un sous-programme supplémentaire qui permet un test très rapide du clavier détectant uniquement l'appui sur la touche BREAK par un CALL 028DH.

La gestion de l'imprimante

La gestion de l'imprimante est faite très différemment sur le modèle 1 et sur le modèle 3 : le premier a l'imprimante implantée en mémoire à l'adresse 37E8H alors que le second se sert du port de sortie ØF8H (qui n'est pas le même que celui du Vidéo-Génie ØFØH !). Les programmes gérant directement l'imprimante, sans passer par les sous-programmes en ROM, ne seront pas compatibles. Par contre le sous-programme permettant d'envoyer un caractère à l'imprimante, CALL ØØ3BH est utilisable sur les deux modèles. De même, pour tester le statut de l'imprimante, on peut utiliser LD A,Ø et CALL ØØ38H, le statut est alors renvoyé dans le registre A.

En supplément, le modèle 3 permet d'envoyer le contenu de l'écran sur imprimante par CALL Ø1D9H, alors que sur un modèle 1, il faut un programme supplémentaire.

Les autres points d'entrée

Il est évident que dans une ROM faisant 12K (modèle 1) ou 14K (modèle 3), il y a énormément de sous-programmes et les tester tous est un gros travail. Les points d'entrée et les sous-programmes suivants sont compatibles entre les deux modèles :

JP ØØØØH : est équivalent à un RESET
JP 1A19H : est le point d'entrée du Basic (READY)
CALL ØØ6ØH : permet de faire une boucle d'attente, le temps étant contrôlé par le registre BC. Les deux modèles travaillant à des vitesses différentes, il se peut que les délais ne soient pas exactement identiques.
Parmi les sous-programmes mathématiques, l'auteur a pu constater que les suivants sont parfaitement compatibles entre les deux modèles :

CALL ØE65H : conversion ASCII --> Binaire
CALL ØFBDH : conversion de FPA1 en ASCII
CALL ØAB1H : conversion de FPA1 en simple précision
CALL Ø7ØBH : (HL) + FPA1 dans FPA1
CALL Ø713H : (RFPA + FPA1) dans FPA1
CALL Ø847H : (RFPA * FPA1) dans FPA1
CALL Ø8A2H : (RFPA / FPA1) dans FPA1
CALL Ø9C2H : (HL) dans RFPA
CALL Ø9CBH : FPA1 dans (HL)
CALL Ø9B1H : (HL) dans FPA1
CALL Ø9BFH : FPA1 dans RFPA
CALL Ø9A4H : FPA1 sur la pile
CALL ØAØCH : comparaison de RFPA et FPA1
CALL Ø982H : change le signe de FPA1
CALL 13E7H : prend la racine carrée de FPA1
CALL Ø977H : prend la valeur absolue de FPA1

(ces sous-programmes sont ceux utilisés par le programme de résolution d'une équation du second degré donné dans le livre THE BOOK - volume I).

Il y a sûrement beaucoup d'autres sous-programmes compatibles (ou non compatibles !) entre les deux modèles. Que les lecteurs nous tiennent informés de leurs découvertes.

5 - CONCLUSION

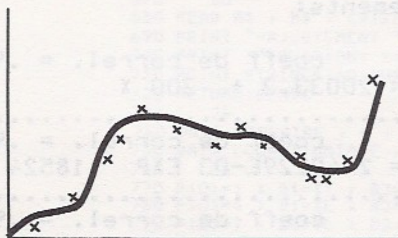
De tout ce qui précède on peut conclure que les deux modèles de TRS-80 ne sont pas 100 % compatibles, mais que beaucoup de programmes du modèle 1 pourront tourner sur un modèle 3.
Si vous voulez acheter des programmes pour un modèle 3, vous devrez vérifier qu'ils peuvent effectivement tourner, les grands fabricants l'indiquant toujours. Dans le doute, n'hésitez pas à demander aux vendeurs de tester le programme sur un modèle 3. Les programmeurs voulant écrire des programmes compatibles devront faire attention à respecter les points d'entrée communs aux deux modèles, et dans tous les cas de les tester sur les deux modèles !!

Jean-Luc Bérardo

Un programme pour mieux prévoir



En mathématiques, on appelle ajustement - ou lissage - la technique consistant à déterminer l'équation théorique d'une courbe à partir d'un nuage de points. En partant des couples de données année/prix du beurre par exemple, on peut en déduire l'allure générale de la courbe correspondant à l'évolution du prix du beurre au fil des années :



De la même manière, on peut mettre en équation l'évolution des loyers, de l'essence ou des salaires.
Comme il est de règle en calcul des statistiques, plus il y aura de couples, meilleure sera la précision.
En général, la plupart de ces phénomènes évolutifs peuvent se classer selon quatre fonctions déterminées par quatre équations :

```
linéaire        : Y = A + B X
exponentielle   : Y = A EXP ( B* X )   (ou A* e ↑ BX)
logarithmique   : Y = A + Ln X         (Ln = log népérien)
puissance       : Y = A X↑B
```

A et B sont les coefficients des équations.

Le programme qui suit détermine, à partir des couples de données, les coefficients de ces quatre équations et précise quelle est la meilleure approximation au moyen de ce que l'on appelle en statistiques le coefficient de corrélation : plus sa valeur est proche de 1, meilleure est l'approximation. Il est ensuite possible de faire une estimation en ne donnant que l'une des valeurs d'un couple (X ou Y), le programme calculant une estimation de l'autre valeur.

Exemple :

En 1970, je gagne 1000 francs, en 1971 : 1200 francs, en 1972 : 1500 francs et en 1973 : 2000 francs.

Combien puis-je espérer gagner en 1974 ? en 1980 ?

(A ce propos, les lecteurs réussissant à faire tenir ce programme sur poquette sont priés de nous en informer : c'est moins encombrant pour s'expliquer avec son supérieur hiérarchique !).

Deux ans plus tard, mes salaires 74 et 75 sont respectivement : 2200 et 2400. Quelle est l'approximation pour 1980 ?

(Voir les réponses dans l'exécution).



Jean-Pierre du Tillet

```
<<< Ajustements de courbes >>>
..............................

Les elements des couples doivent etre > a 0.
Frapper x;y pour introduire un couple;
Rx;y pour retirer le dernier introduit si erreur;
F;F pour arreter l'introduction.

Couple X;Y numero 1 :? 70;1000
Couple X;Y numero 2 :? 71;1200
Couple X;Y numero 3 :? 72;1500
Couple X;Y numero 4 :? 73;2000
Couple X;Y numero 5 :? F;F      <-- fin introduction

Calcul des differents ajustements:

-Ajustement lineaire :              coeff de correl. = .959471
               Equation: Y=-22170 + 330 X
.................................................................
-Ajustement exponentiel :           coeff de correl. = .989663
               Equation: Y= 9.74119E-05 EXP .230273 X
.................................................................
-Ajustement logarithmique :         coeff de correl. = .966299
               Equation: Y=-100148 +  23789.9 LN X
.................................................................
-Ajustement puissance :             coeff de correl. = .996454
               Equation: Y= 2.29823E-28 X[  16.5984
.................................................................
Le meilleur ajustement est puissance .
Suite introduction? N
Estimation X OU Y (ou Introduction ou Stop)? Y
Estimation ajustement puissance
valeur de X? 74
pour X egal 74 , Y egal 2441.7
Estimation X OU Y (ou Introduction ou Stop)? Y
Estimation ajustement puissance
valeur de X? 80
pour X egal 80 , Y egal 8906.2   <-- chouette alors...
Estimation X OU Y (ou Introduction ou Stop)? I
Estimation ajustement puissance
Suite introduction? O
Couple X;Y numero 5 :? 74;2200
Couple X;Y numero 6 :? 75;2400
Couple X;Y numero 7 :? F;F.

Calcul des differents ajustements:

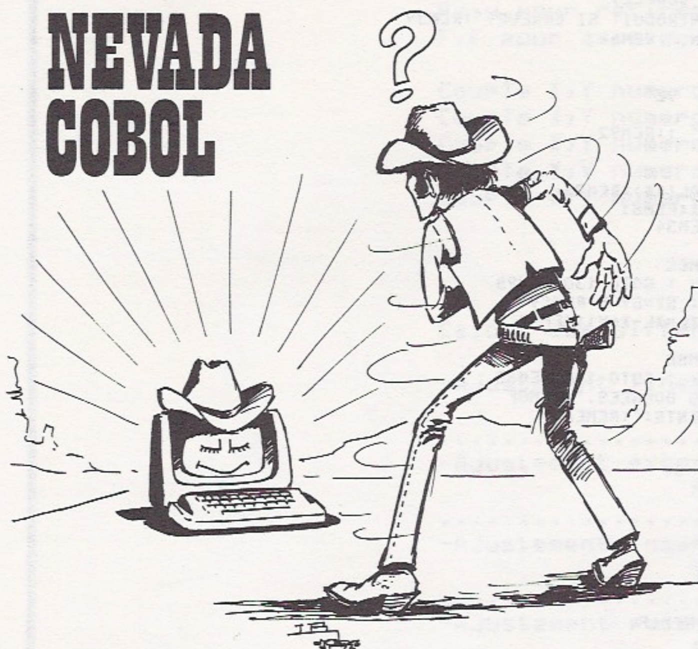
-Ajustement lineaire :              coeff de correl. = .979274
               Equation: Y=-20033.3 +  300 X
.................................................................
-Ajustement exponentiel :           coeff de correl. = .967211
               Equation: Y= 2.40229E-03 EXP  .18524 X
.................................................................
-Ajustement logarithmique :         coeff de correl. = .983783

               Equation: Y=-91808.3 +  21834.7 LN X
.................................................................
-Ajustement puissance :             coeff de correl. = .972735
               Equation: Y= 1.31662E-22 X[  13.4897
.................................................................
Le meilleur ajustement est logarithmique .
Suite introduction? N.

Estimation X OU Y (ou Introduction ou Stop)? Y
Estimation ajustement logarithmique
valeur de X? 80
pour X egal 80 , Y egal 3872.14   <-- BoF !
Estimation X OU Y (ou Introduction ou Stop)? S
READY
>
```

```
10 ' (C) TRACE et l'auteur  F6
20 '  EC
30 CLS : PRINT TAB(15) "<<< AJUSTEMENTS DE COURBES >>>":REM86
40 PRINT TAB(15) STRING$ (30,CHR$(131)) : PRINT:REM6E
50 PRINT "LES ELEMENTS DES COUPLES DOIVENT ETRE > A 0.":REM7A
60 PRINT "FRAPPER X,Y POUR INTRODUIRE UN COUPLE,"
70 PRINT "RX,Y POUR RETIRER LE DERNIER INTRODUIT SI ERREUR,":REM39
80 PRINT "F,F POUR ARRETER L'INTRODUCTION.":REM64
90 PRINT : CLEAR 500 : DIM P(13):REMA4
100 '  9C
110 '             INTRODUCTION DES COUPLES  92
120 '  88
130 PRINT "COUPLE X,Y NUMERO" ; N+1 ; ":" ;:REM92
140 INPUT X$ , Y$ : Y=VAL(Y$):REM9B
150 IF X$="F" THEN 270:REM9D
160 IF LEFT$(X$,1)="R" THEN 220 ELSE X=VAL(X$):REM28
170 N=N+1 : SX=SX+X : SY=SY+Y : X2=X2+X*X:REM61
180 Y2=Y2+Y*Y : M1=LOG(Y) : XL=XL+X*M1:REM34
190 LY=LY+M1 : CY=CY+M1*M1:REMF9
200 M2=LOG(X) : YL=YL+Y*M2 : LX=LX+M2:REMEC
210 CX=CX+M2*M2 : LL=LL+M1*M2 : XY=XY+X*Y : GOTO 130:REM98
220 X=VAL(MID$(X$,2)) : N=N-1 : SX=SX-X : SY=SY-Y:REMC9
230 X2=X2-X*X : Y2=Y2-Y*Y : M1=LOG(Y) : XL=XL-X*M1:REM90
240 LY=LY-M1 : CY=CY-M1*M1:REMC5
250 M2=LOG(X) : YL=YL-Y*M2 : LX=LX-M2:REMB8
260 CX=CX-M2*M2 : LL=LL-M1*M2 : XY=XY-X*Y : GOTO 130:REM62
270 PRINT : PRINT "FIN D'INTRODUCTION DES DONNEES.":REM0F
280 PRINT "CALCUL DES DIFFERENTS AJUSTEMENTS:":REMEA
290 RESTORE : PRINT:REM61
300 '  D3
310 '            CALCUL DES 4 AJUSTEMENTS  C9
320 '  BF
330 FOR AA=1 TO 4:REMBB
340 ON AA GOSUB 770,780,790,800:REMFE
350 GOSUB 680:REM72
360 NEXT AA:REM8E
370 P(0)=0:REMB7
380 FOR A=3 TO 12 STEP 3:REM9A
390   IF P(0)<P(A) THEN P(0)=P(A) : P=A:REM6F
400 NEXT : RESTORE:REM1E
410 FOR A=1 TO P/3:REM8D
420   READ A$ , B$ , C$:REM46
430 NEXT:REMCA
440 '  47
450 '              CONCLUSIONS  3D
460 '  33
470 PRINT "LE MEILLEUR AJUSTEMENT EST " ; A$ ; ".":REM0A
480 INPUT "SUITE INTRODUCTION" ; R$:REM75
490 IF LEFT$(R$,1)="0" THEN 130:REMA4
500 INPUT "ESTIMATION X OU Y (OU INTRODUCTION OU STOP)" ; R$:REMBA
510 IF LEFT$(R$,1)="S" THEN END:REMA0
520 ON P/3 GOSUB 770 , 780 , 790 , 800:REM78
530 PRINT "ESTIMATION AJUSTEMENT " ; A$:REM59
540 IF R$="X" THEN INPUT "VALEUR DE Y" ; VY : GOTO 840:REMD7
550 IF R$="Y" THEN INPUT "VALEUR DE X" ; VX : GOTO 930:REMCE
560 GOTO 480:REMA5
570 '  C4
580 '             CALCUL DES PARAMETRES  BA
590 '  B0
600 NB=B1-B4*A1/N : B=NB/(B3-B4*B4/N):REMDD
610 R=B*NB/(R1-A1*A1/N):REM91
620 A=(A1-B*B4)/N : IF P(0)=4 OR P(0)=10 THEN A=EXP(A):REM77
630 P(P(0))=B : P(P(0)+1)=A : P(P(0)+2)=R:REM6D
640 RETURN:REMEC
650 '  74
660 '            AFFICHAGE DE L'AJUSTEMENT  6A
670 '  60
680 READ A$ , B$ , C$:REM41
690 PRINT "-AJUSTEMENT " ; A$ ; ":" , "COEFF DE CORREL. =" ; R:REMB5
700 PRINT ,"EQUATION: Y=" ; A ; B$ ; B ; C$:REMAE
710 PRINT STRING$(60,CHR$(131)):REM02
720 RETURN:REM9C
730 '  24
740 DATA LINEAIRE , + , X , EXPONENTIEL , EXP , X:REM5A
750 DATA LOGARITHMIQUE , + , LN X , PUISSANCE , X[ ,:REM56
760 '  06
770 P(0)=1 : B1=XY : B3=X2 : B4=SX : A1=SY : R1=Y2 : GOSUB 600 : RETURN:REM3C
780 P(0)=4 : B1=XL : B3=X2 : B4=SX : A1=LY : R1=CY : GOSUB 600 : RETURN:REM32
790 P(0)=7 : B1=YL : B3=CX : B4=LX : A1=SY : R1=Y2 : GOSUB 600 : RETURN:REM24
800 P(0)=10 : B1=LL : B3=CX : B4=LX : A1=LY : R1=CY : GOSUB 600 : RETURN:REMF3
810 '  D3
820 '            ESTIMATION DE X  C9
830 '  BF
840 ON P/3 GOSUB 860 , 870 , 880 , 890:REM2E
850 PRINT "POUR Y EGAL" ; VY ; ", X EGAL" ; VX : GOTO 500:REMB2
860 VX = (VY - A) / B : RETURN:REM31
870 VX = LOG (VY / A) / B : RETURN:REM46
880 VX = EXP ( (VY-A) / B ) : RETURN:REMEC
890 VX = (VY/A) [ (1/B) : RETURN:REMBE
900 '  79
910 '            ESTIMATION DE Y  6F
920 '  65
930 ON P/3 GOSUB 950 , 960 , 970 , 980:REMD4
940 PRINT "POUR X EGAL" ; VX ; ", Y EGAL" ; VY : GOTO 500:REM58
950 VY = A + B * VX : RETURN:REM2A
960 VY = A * EXP (B * VX) : RETURN:REMED
970 VY = A + B * LOG (VX) : RETURN:REME6
980 VY = A * (VX [ B) : RETURN:REMB7
```

NEVADA COBOL



Le COBOL, en tant que langage de programmation le plus utilisé en gestion, est crédité d'un certain nombre d'avantages, qui sont notamment la transportabilité, l'adaptation spécifique au traitement de problèmes de gestion, la bonne lisibilité des programmes qui facilite la maintenance.

. Transportabilité

Le COBOL, comme le BASIC, et malgré sa normalisation, possède un certain nombre de "dialectes". Son évolution, à travers différentes versions, à amené des modifications pas toujours compatibles entre elles.

Certaines particularités du langage sont liées au type de machine utilisé ou à l'environnement périphérique. De plus, le COBOL est constitué d'un certain nombre de modules, qui pour une version donnée, peuvent être présents ou non.

Ceci montre qu'en fait la portabilité complète du COBOL est loin d'être assurée, et c'est surestimer les possibilités de ce langage, que d'affirmer que son installation sur les ordinateurs individuels va automatiquement ouvrir la porte à un nombre considérable de programmes existants. Dans le meilleur des cas un effort de conversion important restera à faire, sans compter que la majorité des programmes COBOL existant actuellement, n'a pas été écrite pour des systèmes interactifs ; de même, les programmes plus récents gèrent des terminaux et non pas un clavier/écran faisant directement partie de l'ordinateur.

Il faut souhaiter le développement d'un standard COBOL pour les ordinateurs individuels afin de pouvoir compter sur les avantages d'une certaine transportabilité des programmes.

. Adaptation aux problèmes de gestion

Sur ce plan le COBOL a indéniablement des atouts. Quelles sont les facilités attendues pour réaliser, dans de bonnes conditions, des programmes de gestion ?

Essentiellement une présentation des résultats qui soit facilement lisible et compréhensible pour l'utilisateur non informaticien ; c'est-à-dire, des fonctions de mise en page et de contrôle exact du format des valeurs numériques sont d'une importance primordiale.

Ici le COBOL montre tous ses avantages. La longueur, le cadrage et le format de chaque rubrique sont parfaitement et facilement contrôlables, ainsi que, pour les valeurs numériques, le nombre de décimales entières et fractionnaires. La précision des résultats est absolue. Il n'existe pas d'arrondis approximatifs introduits par des parties fractionnaires que vous n'avez pas demandées et qu'il faudra ajuster au fur et à mesure, comme c'est le cas en BASIC. Les formats de sortie à l'écran ou en impression sont facilement composés et modifiés.

Alors que les sorties destinées à l'utilisateur peuvent prendre les formes les plus clairement lisibles, il est facile d'utiliser des formats internes qui demandent moins de place en mémoire et sur les supports magnétiques et qui accélèrent les opérations de calcul et d'entrée/sortie. Ceci donne au COBOL une puissance de calcul appréciable au moins en ce qui concerne les quatre opérations de base. Pour le travail sur les supports d'accès direct (disques, disquettes, etc ...) le COBOL propose des types d'accès très puissants (Séquentiel Indexé, Random, Gestion de Bases de Données, types d'accès mixtes). Différents formats d'enregistrement et de groupage d'enregistrements (fixe, variable) font du COBOL un puissant outil de gestion de fichiers.

. Lisibilité de programmes

Comparé au PASCAL, le COBOL a une structure bien moins contraignante. Il est donc possible d'écrire, en y mettant une certaine application, des programmes tout à fait hermétiques, surtout si ces programmes dépassent plusieurs milliers de lignes, ce qui n'est pas rare, le COBOL étant bien plus "verbeux" que le BASIC.

Il existe néanmoins une structure du COBOL qui sépare un programme en quatre divisions comportant chacune des sections distinctes. Toute information se rapportant à une division ou section ne peut être écrite qu'entre l'entête de la division (ou section) et l'entête suivante.

Ainsi sont séparées la description de l'environnement (l'ordinateur et les périphériques à utiliser), la description des fichiers, y compris les formats de l'imprimante et de l'écran, la réservation des zones de travail, des zones de liaison avec des sous-programmes, et la procédure, c'est-à-dire, les instructions COBOL proprement dites.

Cette construction et la syntaxe du COBOL, proche de l'anglais, renforcée par la possibilité d'utiliser des notes, des noms de section, de paragraphes et de rubriques parlants, facilitent la compréhension des programmes. De l'absence d'une structure obligatoire à l'intérieur de la partie traitement (Procédure Division) découle la possibilité d'utiliser diverses méthodes destinées à faciliter la programmation, la maintenance et à améliorer la lisibilité (Programmation structurée, LCP-Warnier, Programmation Modulaire, etc ...).

. Adaptation aux ordinateurs individuels

Du fait que le COBOL est un langage très précis, qui ne suppose aucune option prise par défaut par le compilateur, il est évident que le moindre programme

demande un nombre de lignes d'instructions nettement plus important que le BASIC. Il faut en tenir compte pour la saisie des programmes. Classiquement ce travail n'est pas fait par les programmeurs, qui écrivent leurs programmes sur des feuilles de codage destinées à des opératrices de saisie. Pour l'utilisateur d'un ordinateur individuel qui fait tout lui même, il s'agit de savoir si les mérites du COBOL valent pour lui ce travail supplémentaire.

L'avantage du COBOL sera surtout sensible pour ceux qui ont déjà une certaine expérience du COBOL, c'est à-dire la grande majorité des informaticiens actuels.

. Les caractéristiques du Nevada COBOL

Ce compilateur, d'un prix relativement abordable (150 $ aux USA, environ 1500 F en France) fonctionne sur le Modèle 1, sous CP/M exclusivement. Il faut donc acheter ce système en plus du compilateur. Pour les utilisateurs de TRS-DOS, LDOS, ou autres NEWDOS, cela demande un sérieux effort d'adaptation et le renoncement à certaines facilités de ces systèmes.

En ce qui concerne le Nevada COBOL proprement dit, il faut tout de suite dire qu'il souffre de lacunes sévères. La première de ces lacunes étant l'absence d'un éditeur de programmes. Si vous ne disposez pas d'un éditeur ou d'un programme de traitement de textes fonctionnant sous CP/M, il ne vous reste que la fonction EDIT du CP/M qui est peu pratique à utiliser.

En ce qui concerne les possibilités du langage lui-même, il y a aussi des lacunes importantes. La première à mon avis, l'absence d'une possibilité de décrire des grilles d'écran. Tous les dialogues avec celui-ci se font à travers les verbes DISPLAY et ACCEPT qui font aller à chaque fois à la ligne. De ce fait, le dialogue utilisateur-machine, ne pourra être que très rudimentaire.

Il n'y a par ailleurs, aucun verbe d'accès direct ni d'accès séquentiel indexé, pas d'ordres de tri, pas de LINKAGE SECTION, pas de tests imbriqués ni même de ELSE, pas de NOM CONDITION, le PERFORM n'existe que dans sa forme la plus rudimentaire, et beaucoup d'autres fonctions des plus puissantes du COBOL sont absentes (MOVE CORRESPONDING, COMPUTE, SEARCH, Report Writer, etc ...).

Le manque le plus important peut-être, c'est l'absence des verbes CALL ou ENTER qui permettent d'appeler des routines dans d'autres langages. La possibilité d'incorporer des routines Assembleur aurait permis de pallier à certaines des faiblesses du Nevada COBOL.

. L'intérêt du Nevada COBOL

Alors quel peut-être l'intérêt d'utiliser ce compilateur ?

D'abord son prix, qui le place très loin de ses concurrents.

Il peut, dans un premier temps, servir à découvrir le COBOL. Il contient en effet les éléments de base de ce langage, utiles à la compréhension des principes du COBOL, tels que les descriptions de fichiers et de zones, et les ordres de traitement élémentaires.

Les fonctions remplies par ce compilateur, permettent la réalisation de programmes d'édition et de gestion de fichiers dans d'excellentes conditions, aussi longtemps qu'il n'est pas fait appel à un dialogue important avec l'opérateur. Parmi les applications envisageables se situent toutes sortes d'utilitaires, tels que tris et fusions.

Toutefois, pour construire une application entière sur ce langage, il serait souhaitable de disposer d'un COBOL plus complet.

Wolfgang Lauter.

matériel à l'essai

CP/M pour votre modèle 1

* Qu'est-ce que CP/M

Un ordinateur sans système d'exploitation, vous le savez, est comme une voiture sans chauffeur. En effet, le système d'exploitation (appellé Disk Operating System ou DOS en anglais) permet à l'utilisateur d'effectuer par des commandes simples tout un ensemble d'opérations complexes, par exemple :

- gestion de l'espace disque
- gestion des entrées/sorties disque
- communication avec les autres organes périphériques

En ce qui concerne le TRS-80, il existe pour lui deux familles de DOS : d'une part TRSDOS de Radio Shack et ses dérivés, tel que NEWDOS80 et LDOS, d'autre part CP/M. Notre dernier (et premier !) numéro vous présentait l'utilisation de CP/M sur un modèle 2, nous nous intéressons ici aux modèles 1 et 3.
CP/M (Control Program Monitor) a été développé par Digital Research pour le microprocesseur 8080 d'Intel mais depuis, il a été adapté pour tous les microprocesseurs compatibles avec le 8080, donc notamment le Z80 qui équipe votre TRS-80.

* Portabilité des programmes

Un des aspects qui rend CP/M particulièrement intéressant est la possibilité d'échanger des programmes entre ordinateurs de marques différentes. D'abord parce que le support de programme est tout à fait normalisé (enregistrement sur disquette en format IBM 3740), et ensuite parce que les entrées/sorties (écran, clavier, disquettes etc ...) utilisent des sous-programmes contenus dans CP/M lui-même (la partie BIOS ou Basic Input Output System, qui lui, évidemment, est particulier pour chaque type de machine).

CP/M existe depuis 6 ans, et à l'heure actuelle on peut le considérer comme parfaitement rodé et bien au point. Plus de 100 compagnies offrent des programmes de qualité excellente dans tous les domaines, tel que comptabilité, gestion de stocks, traitement de texte, jeux, etc ... De plus, CP/M a été choisi par de nombreux constructeurs d'ordinateurs, notamment IBM pour son Personal Computer (dans une version légèrement differénte, il est vrai).

* Le "Package" CP/M pour TRS-80

CP/M est disponible sur TRS-80 auprès de la firme américaine OMIKRON Systems, qui a adapté le BIOS pour les périphériques du TRS et a également développé un circuit électronique modifiant l'adressage de la mémoire vive quand le TRS travaille sous CP/M.

Comme la carte Mapper II, qui permet l'utilisation des disquettes 8" sur le TRS (voir banc d'essai dans le premier numéro de TRACE), le package CP/M n'est malheureusement pas disponible en France. Je l'ai donc commandé directement chez OMIKRON, et c'est après trois mois d'attente que le paquet est arrivé. Il contenait :

- un petit circuit imprimé (la carte MAPPER I)
- une disquette 5" CP/M version 1.4
- un manuel d'utilisation OMIKRON
- l'ensemble de la documentation CP/M (6 manuels)

Le système CP/M fourni était la version 1.4, mais OMIKRON propose aujourd'hui également la version 2.2, plus récente.

Pour utiliser le package CP/M, il faut un TRS-80 modèle 1 plus l'extension d'interface avec au moins 16K de mémoire, ainsi qu'un lecteur de disquettes 5". Il convient de signaler, que le système est "fait sur mesure" pour la taille de mémoire disponible, et aussi pour la configuration des lecteurs de disquettes, ces informations étant précisées lors de la commande.

A titre d'exemple, mon système était configuré pour 48K de mémoire, les drives A et B comme des 5", et les drives C et D comme des 8".

* Installation et mise en oeuvre

L'installation du circuit imprimé est simple. Elle ne nécessite ni manipulation de fer à souder ni coupure de pistes. Tout ce qu'il y a à faire, c'est de retirer le Z80 (doucement quand même - les pattes sont fragiles !), insérer à la place la carte Mapper (qui contient deux rangées de broches qui entrent dans le support du Z80), et enfin remettre en place le Z80 dans le support prévu à cet effet sur le Mapper.

Maintenant on peut mettre le TRS-80 sous tension. Si vous avez bien suivi les conseils, le message suivant s'affiche sur votre écran :

T=TRS-80
C=CP/M

En appuyant sur T, on a un TRS-80 tout à fait normal - la carte Mapper est complètement transparente pour l'utilisateur - et la machine lit le système TRSDOS (ou NEWDOS80 ou LDOS) placé dans le drive :0.
En appuyant sur C, la machine charge CP/M depuis le drive A et attend une commande de l'utilisateur.

En plus des modules CP/M standards tel que assembleur, éditeur de texte, programme PIP (Peripheral Interchange Program) pour échange entre les différentes unités périphériques, la disquette système contient un certain nombre de programmes utilitaires, développés par OMIKRON :

- test des lecteurs de disquettes

- test de mémoire
- formattage de disquettes 5" et 8" (format IBM 3740)
- duplication de disquettes 5" et 8"
- driver série pour la porte RS-232C

Le manuel d'utilisation sur 22 pages (en anglais) détaille l'installation de la carte Mapper et décrit tous les utilitaires. La documentation CP/M, d'origine Digital Research, comporte six manuels :

- An Introduction to CP/M Features and Facilities
- CP/M System Alteration Guide
- CP/M Dynamic Debugging Tool (DDT)
- CP/M Assembler (ASM)
- ED: A Context Editor for the CP/M Disk System
- CP/M Interface Guide

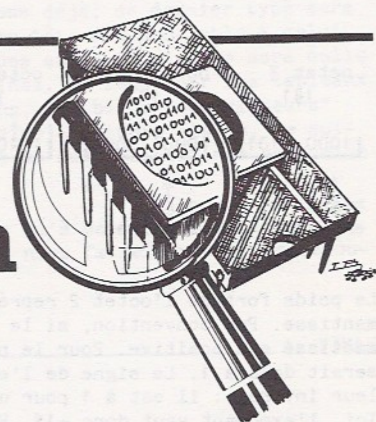
Le système CP/M est fourni sans language de haut niveau tel que le BASIC, mais il existe plusieurs interprêteurs et compilateurs qui travaillent sous CP/M, notamment MBASIC de Microsoft, qui ressemble beaucoup au Disk BASIC du TRS-80, et CBASIC qui est un BASIC semi-compilé. Pour vérifier la compatibilité entre mon TRS-80, travaillant en CP/M, et un autre ordinateur (en l'occurrence une machine de traitement de texte équipée d'un INTEL 8080 et de deux disquettes 8", disposant également du CP/M), j'ai essayé de lire sur le TRS une disquette 8" contenant le compilateur CBASIC de l'autre machine. Compilation et exécution se sont déroulées sans aucun problème !

Le prix était de 199 dollars US pour l'ensemble carte Mapper I, disquette CP/M et documentation, plus 25 dollars de port et emballage, soit un total de 224 dollars. L'achat m'a coûté en tout 1275 francs en tenant compte des frais de douane, de timbres et de TVA. Le dollar m'est donc revenu à 5,70 francs, soit 26 % de plus que le cours officiel, qui à cette époque, était de 4,53 francs. Eh oui, nous sommes bien loin des prix prohibitifs que l'on constate souvent chez les importateurs/revendeurs français ! L'importation personnelle peut être payante !

Steen Dalby Pedersen

technique

Les variables et leur représentation en mémoire



Avec un langage évolué comme le BASIC, la manipulation des variables est chose aisée, ne serait-ce que par le simple fait que nous n'avons pas à nous soucier de leur disposition en mémoire centrale. Nous utilisons ces variables comme de simples "boîtes à lettres" qui répondent automatiquement à l'appel de leur nom : A=3, Y=X-A... Bien pratique hein ?. Il faut pourtant savoir que l'interprêteur lui, ne s'amuse pas du tout pendant ce temps-là !

Nous pourrions bien sur en rester là, et nos programmes BASIC continueraient malgré tout à fonctionner correctement. Toutefois, il peut être avantageux dans certaines circonstances, d'exploiter les variables par une routine en langage machine appelée par USR. Dans ce cas, il est nécessaire de connaître la représentation physique de ces variables en mémoire centrale. Les fonctions BASIC PEEK et VARPTR sont les clés qui vont nous permettre cette exploration.

Rappelons que la fonction VARPTR (<nom de variable>) fournit une valeur d'adresse qui permettra de retrouver la variable en mémoire. PEEK (<valeur d'adresse>) retourne l'octet mémoire rangé à cette adresse. Attention : si la valeur d'adresse est supérieure à 32767, il convient de lui retirer 65536 (PEEK, tout comme POKE, n'accepte que des valeurs ENTIERES, donc comprises entre -32768 et +32767).

Variables numériques

On trouve les variables numériques sous trois formes dans le BASIC du TRS-80 : entières (%), flottantes simple précision (!) et flottantes double précision (#).

Le plus simple est de prendre un exemple de programme et d'étudier les résultats fournis :

```
10 AB% = 4660         ' variable entière
20 N=0                ' réservation variable de boucle
30 A = VARPTR (AB%)   ' pointe AB% en mémoire
40 FOR N=-3 TO 7
50 PRINT PEEK (A+N);
60 NEXT : PRINT
```

ATTENTION: la ligne 20 est absolument nécessaire (voir TRACE n°1, page 11).

Lançons ce programme et notons les valeurs obtenues. La même opération sera effectuée en remplaçant successivement AB% aux lignes 10 et 30 par AB! puis par AB#. Voici le tableau récapitulatif des résultats :

	PEEK (A+N)										
N	-3	-2	-1	0	+1	+2	+3	+4	+5	+6	+7
AB%	2	66	65	52	18	//////	//////	//////	//////	//////	//////
AB!	4	66	65	0	160	17	141	//////	//////	//////	//////
AB#	8	66	65	0	0	0	0	0	160	17	141

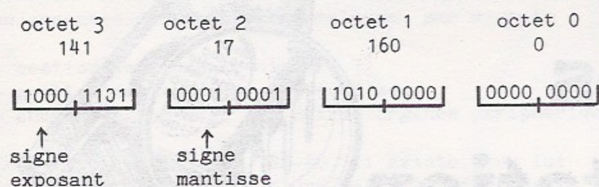
La zone hachurée n'est pas significative.
Le VARPTR de chaque variable pointe à partir de l'octet pour lequel N=0 et s'étend jusqu'à N=7. Nous appelons cette zone "zone utile". La zone située à gauche (N=-3 à N=-1) sera baptisée "en-tête" de la variable. La zone utile est la seule qui soit vraiment importante pour le programmeur, mais pour le petit curieux, l'en-tête contient quelques renseignements intéressants. Ainsi, l'octet N=-3 contient la longueur de la zone utile, alors que les octets N=-2 et N=-1 reçoivent les deux premiers caractères ASCII du nom de la variable : 66 pour "B" et 65 pour "A".

Comme toute représentation MEMOIRE, ces octets sont inversés par rapport à la progression des adresses. La "zone utile" contient la valeur de la variable, sous une forme simple dans le cas de la variable entière (52 + 256 * 18 = 4660), un peu moins simple dans le cas des variables flottantes. Ces deux dernières représentations méritent d'être approfondies. A première vue, il ne semble pas évident que 0, 160, 141 ait un rapport quelconque avec 4660...

Pour mieux comprendre, il est nécessaire de placer les octets dans le sens inverse. Commençons par la variable simple précision :

141 17 160 0

C'est déjà plus clair, non ?
Le nombre 4660 est ici représenté sous la forme flottante normalisée, 141 étant l'exposant et 17 160 0 les trois octets de la mantisse. Mais il faut passer par la représentation binaire pour que cela soit plus clair encore.



Le poids fort de l'octet 2 représente le signe de la mantisse. Par convention, si le bit est à zéro, la mantisse est positive. Pour le nombre -4660, ce bit serait donc à 1. Le signe de l'exposant prend la valeur inverse : il est à 1 pour un exposant positif. Ici, l'exposant vaut donc +13. En ce qui concerne la mantisse, deux cas peuvent se présenter : soit elle est représentée dans son intégralité par les octets 2, 1 et 0, soit on part du principe très évident que le premier bit (celui de gauche) ayant toujours la valeur 1, il est inutile de le faire figurer dans la mantisse (on gagne ainsi la place d'un bit). Dans ce dernier cas qui sera le notre, il faudra donc ajouter implicitement un bit 1 devant notre valeur de mantisse, lors de la conversion. Ainsi, pour retrouver notre valeur 4660, il suffit de prendre la mantisse sans son signe, de la faire précéder d'un bit 1 supplémentaire, comme ceci :

1 0001 1010 0000 0000 0000

puis de placer un "point binaire" après le treizième bit (valeur de l'exposant sans son signe) en partant de la gauche :

1 0010 0011 0100 . 0000 0000 0000

Ce nouveau nombre, converti en hexadécimal nous donne 1234H, soit 4660 en décimal et c'est bien celui que nous cherchions !
Le mode opératoire est le même pour les variables flottantes double précision, mais s'effectue sur huit octets au lieu de quatre. La représentation des nombres à exposant négatif est plus complexe et nous ne l'exposerons pas ici.

Variables de chaines alphanumériques

Nous conserverons le petit programme décrit au début de cet article, en remplaçant AB% par AB$ en ligne 30, et AB%=4660 par AB$="1234" en ligne 10. Observons les résultats :

N	PEEK (A+N) -3	-2	-1	0	+1	+2	+3	+4	+5	+6	+7
AB$	3	66	65	4	81	106	//////	//////	//////	//////	//////

Si l'en-tête a la même signification que pour les variables numériques, la zone utile, par contre, conduit à une interprétation différente : l'octet 0 nous informe sur la longueur de la chaîne alphanumérique, et les octets +1 et +2 contiennent l'adresse de cette chaîne (dans notre cas, ce sera 81 + 256 * 106). Lorsque la chaîne est définie clairement dans le programme (ex : AB$="1234"), cette adresse pointe dans le programme lui-même. Par contre, si la chaîne est construite dynamiquement, son adresse pointera dans l'espace réservé aux chaînes alphanumériques situé en fond de mémoire (dimensionné par CLEAR). Dans ce cas, et si vous avez plus de 16K de mémoire RAM, cette adresse sera probablement supérieure à 32767 et devra être convertie en valeur entière (voir ci-dessous).

Pour accèder à cette chaîne, on peut utiliser un programme de ce style :

```
100 L=PEEK (A)                           'longueur de la chaîne
110 CH = PEEK (A+1) + 256 * PEEK (A+2)
                                         'adresse chaîne
115 IF CH > 32767 THEN CH=CH-65536 'valeur entière
120 FOR N=0 TO L-1
130 PRINT PEEK (CH+N);
140 NEXT : PRINT
```

On peut aussi faire :

```
130 PRINT CHR$ ( PEEK (CH+N) );
```

ce qui donnera le même résultat que PRINT AB$... mais en nettement plus compliqué!

Tableaux numériques

Pour les tableaux, nous retrouvons une structure similaire à celle des variables, formée d'une zone "en-tête" et d'une zone utile. L'en-tête est cette fois composé de 5 octets plus 2 octets par dimension). Pour cette raison, il est difficile de "remonter" l'en-tête en partant de l'adresse donnée par le VARPTR, si l'on ne connaît pas à priori le nombre de dimensions du tableau (toutefois en pratique, celui-ci est toujours connu du programmeur au moment de l'appel du VARPTR). Il faut noter que l'en-tête n'est significatif que lorsque le VARPTR pointe sur le premier élément du tableau.

Reprenons le programme que nous avions utilisé pour les variables numériques et remplaçons AB% par AB% (0) puis par AB! (0), et enfin par AB# (0). La boucle devra s'étendre de N=-7 à N=7 en ligne 40. L'en-tête nous donne :

N	PEEK (A+N) -7	-6	-5	-4	-3	-2	-1
AB%(0)	66	65	25	0	1	11	0
AB!(0)	66	65	47	0	1	11	0
AB#(0)	66	65	91	0	1	11	0

Les octets -7 et -6 nous sont devenus familiers : ils contiennent le nom du tableau. Les octets -5 et -4 donnent une longueur sur 16 bits permettant de con-

naître la fin du tableau (dans notre cas, elle se situera 25 octets plus loin). Cette information peut être calculée par la formule :

LG = longueur zone utile + 1 + 2 * nombre de dimensions

L'octet -3 contient le nombre de dimensions du tableau (1 dans notre cas) et les octets -2 et -1 donnent le nombre d'éléments alloués à chaque dimension. Notre tableau n'ayant pas été dimensionné, un DIM AB%(10) implicite nous est imposé par l'interprêteur, ce qui autorise l'emploi des éléments 0 à 10, soit 11 éléments au total. Si nous ajoutons à notre tableau une ou plusieurs dimensions, les octets -3 à -7 seront décalés vers la gauche d'autant de fois deux octets qu'il y a de dimensions supplémentaires ... Un exemple ?

Modifions notre programme ainsi :

```
5 DIM AB%(1,2)
10 AB%(0,0) = 4660
30 A = VARPTR (AB%(0,0))
40 FOR N=-9 TO 7
```

Nous obtiendrons l'en-tête :

N	PEEK (A+N) -9	-8	-7	-6	-5	-4	-3	-2	-1
AB%(0,0)	66	65	17	0	2	3	0	2	0

Les octets -7 et -6 donnent la longueur totale du tableau (soit pour un tableau 2 dimensions de 2*3 éléments : 17 octets)..

L'octet -5 nous informe cette fois que le tableau possède 2 dimensions, et les octets -4, -3 et -2, -1 donnent le nombre d'éléments dans chaque dimension spécifiée par le DIM.

La zone utile commençant à partir de l'octet N=0 est semblable pour les trois types de tableaux à celle des variables numériques. Dans le cas d'un tableau à plusieurs dimensions, les éléments se suivent consécutivement colonne par colonne, sous la forme :

1,c 1+1,c 1+2,c... 1,c+1 1+1,c+1... etc.

A chaque fois que nous demandons le VARPTR d'un élément de tableau multi-dimensionnel (A%(3,5,12) par exemple, l'interprêteur doit consulter l'en-tête de ce tableau afin de nous fournir l'adresse correcte de la zone utile correspondante.

Tableaux de chaînes alphanumériques

Avec ce que nous savons déjà, ce dernier type sera facile à décrire. L'en-tête est semblable à celui des tableaux numériques et la zone utile sera celle des variables de chaînes. De même pour les tableaux à plusieurs dimensions, les blocs "descripteurs" (longueur/adresse chaîne) de trois octets, se suivront consécutivement en mémoire, colonne par colonne.

Il ne vous reste plus qu'à mettre en pratique ces informations ... et à nous faire part de vos découvertes !

A. PINAUD

utilitaire

Des outils pour votre programmation

Dès que l'on commence à s'attaquer à l'écriture en BASIC de programmes importants, on se trouve tout naturellement assez rapidement bloqué par les capacités de la machine. Et comme on n'a guère le choix, il faut commencer à ruser : ruser avec le BASIC, en maîtrisant ses défauts pour les transformer en avantages ; ruser avec les bonnes règles de la programmation, afin d'obtenir un programme aussi court que possible, même si les lignes très longues ne sont compatibles ni avec l'esthétique, ni avec la sagesse des règles en vigueur ; ruser avec les performances, en écrivant en langage machine certains passages délicats.
Autant de techniques qui ne sont faciles ni à apprendre (elles sont rarement expliquées), ni à maîtriser (lorsque'on marche en terrain miné, il y a toujours quelques dangers !).
Afin de vous aider dans cette délicate connaissance, nous vous présentons une série d'articles. Ceux-ci vous montreront quelques-uns des outils et des méthodes "rusés" que vous pouvez utiliser pour vos programmes. Et, "last but not the least", nous mettrons en oeuvre ces méthodes pour en tirer un splendide outil de mise au point.

Que pensez-vous d'un programme moniteur d'aide à la mise au point écrit en assembleur et chargé en mémoire par un CLOAD ou un LOAD ? Programme capable de se reloger, de s'auto-protéger et de s'escamoter en mémoire, quelle que soit la configuration du système (16, 32, 48 K, cassette ou disquette) ?

Nous allons vous présenter les techniques "tabou" qui ont été employées dans la réalisation d'un utilitaire nommé BASPY (espion BASIC), et vous fournir la liste complète de ce programme qui permet sous BASIC

- le dump mémoire sous forme hexa et ASCII,
- les conversions hexadécimal / décimal,
- l'édition mémoire (lecture / écriture),
- la recherche en mémoire d'une configuration exprimée en hexadécimal ou en ASCII,
- le retour au BASIC au point interrompu.

De par sa constitution, BASPY est auto-protégé car placé en mémoire avant la zone BASIC, ce qui permet de laisser libre l'espace s'étendant jusqu'en fin de mémoire (pas de MEMORY SIZE à définir), et d'éviter ainsi la multiplicité des versions liées aux tailles mémoire ou aux organes périphériques (cassette ou disquette).

En version disque, le programme chargé par BASIC s'implante automatiquement en fonction du nombre de zones tampons réservées aux fichiers et il est compatible avec les différentes versions actuelles de TRSDOS, NEWDOS, NEWDOS80 et LDOS.

Ce programme peut coexister en permanence avec l'interprêteur BASIC, et il facilite la mise au point des programmes en fournissant un lien DIRECT avec la mémoire. Il peut de plus se révéler être un outil pédagogique permettant de mieux saisir les échanges entre le BASIC et la machine.

OUTIL N° 1 : DEPLACER LA ZONE BASIC

Sous BASIC, la meilleure place pour une routine, la place la plus sûre, celle où elle ne risque pas de se faire "écraser", ce n'est pas en fond de mémoire (quel fond d'abord ? 16K ? 32K ? 48K ?), c'est AVANT la zone BASIC.
Que trouve-t-on dans l'espace s'étendant entre l'adresse 4000H (début de la mémoire vive) et la zone BASIC (l'endroit où se chargera tout programme écrit en BASIC) ?. Cette zone est "élastique" ... On y trouve d'abord les talbes et les pointeurs réservés à l'interprêteur. En version cassette, la zone BASIC commence là, aux alentours de 42E9H (17129). En version disque, nous trouvons d'abord le système d'exploitation (SED, DOS pour les anglicistes) et les extensions du DISK-BASIC. Viennent ensuite les zones de manoeuvre réservées aux fichiers disquettes : autant de zones de 290 octets que vous avez réservé de fichiers en réponse à HOW MANY FILES. Enfin, arrive la zone BASIC, zone dans laquelle sera chargée le programme écrit en BASIC.
L'adresse de cette zone est fort heureusement conservée bien au frais dans un endroit immuable : 40A4H et 40A5H, (16548 et 16549 en décimal), qui en contiennent respectivement les poids faibles et les poids forts.

LE DEBUT DE LA ZONE BASIC EST POINTE PAR 40A4H.

L'OCTET PRECEDANT LA ZONE BASIC DOIT TOUJOURS ETRE ZERO.

Et la fin de la zone BASIC, maintenant ? Comment peut-on la connaître ?
Facile ! Le début de la zone des VARIABLES utilisées dans un programme BASIC est pointé par les adresses 40F9H et 40FAH. C'est également un endroit immuable, et aussi une adresse à se rappeler. Lorsque l'on sait que cette zone suit immédiatement la zone BASIC ...

LA FIN DE LA ZONE BASIC EST POINTEE PAR 40F9H.

Maintenant, nous savons tout ce qui est nécessaire de savoir pour déplacer la zone BASIC : il faut augmenter la valeur contenue aux adresses 40A4H et 40A5H (début de la zone), ainsi que celle contenue aux adresses 40F9H et 40FAH (fin), en n'oubliant pas de remettre à zéro l'octet précédent le nouveau début de la zone. L'espace ainsi sauté pourra être utilisé par exemple pour loger une routine binaire.

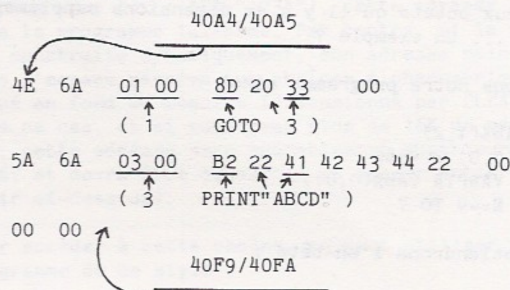
Mais avant de pratiquer un peu avec ce genre d'exercice, examinons la structure de la zone BASIC proprement dite.
En mémoire, chaque ligne BASIC ne ressemble que de fort loin à celle que l'on peut lire sur une liste de programme. Elle est composée de 4 champs :

- un "lien de chaîne" sur 2 octets, indiquant l'adresse de la ligne suivante (poids faibles en tête), et utilisé par l'interprêteur pour une recherche rapide, au travers du programme.
- le numéro de la ligne sur 2 octets, code en binaire pour tenir moins de place, (poids faibles en tête).
- la suite des instructions BASIC, les "mots-clés" de l'interprêteur étant codés sur un seul octet (l'ordre PRINT par exemple, sera codé B2H en hexa).
- un octet zéro indiquant la fin de la ligne.

Un lien de chaîne de valeur 0 (sur deux octets) indique la fin du programme.
Ainsi, si l'on écrit :

```
1 GOTO 3
3 PRINT"ABCD"
```

nous retrouverons ce petit programme dans la zone BASIC sous cette forme :



(la valeur du lien de chaîne peut varier selon les configurations).
Dans cet exemple, 40A4H pointera sur l'octet "4E" et 40F9H pointera après le lien de chaîne nul.

Voici maintenant le petit programme (seulement 2 lignes) qui vous permettra de déplacer la zone BASIC.
ATTENTION : la seconde ligne ne doit pas être coupée.

```
10 INPUT "de combien d'octets voulez-vous déplacer
le BASIC";D
20 A=PEEK(16548)+256*PEEK(16549):A=A+D:POKE16548,
(A AND 255):POKE16549,A/256:POKE A-1,0:POKEA,0:POKEA
+1.0:NEW
```

Pour vérifier la chose, et <u>avant</u> de charger le programme, frappez d'abord :

```
NEW
PRINT MEM
```

et notez la valeur affichée.
Chargez ensuite le programme et lancez-le.
Répondez à la question avec la valeur 200, par exemple.
Après exécution, faire de nouveau :

```
PRINT MEM
```

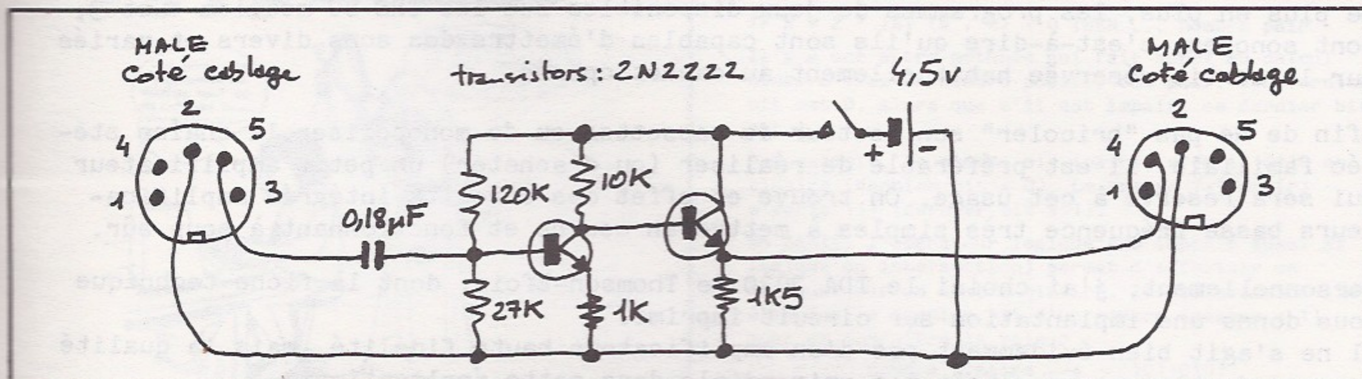
Vous devez retrouvez la valeur initiale diminuée de 200, ce qui prouve bien que la zone BASIC à "avancée" de 200 octets. Pour revenir à l'ancienne valeur, relancer le programme (après l'avoir chargé) et frapper -200 ...

Remarques sur le programme :
En ligne 20, on lit tout d'abord l'adresse contenue en 40A4 et 40A5 (16548 et 16549). Pour reconstituer une valeur "16 bits", il est nécessaire de multiplier les poids forts par 256 et d'y ajouter les poids faibles. Cette valeur est ensuite stockée dans la variable A, puis est augmentée de celle de la variable D (valeur de déplacement). On modifie alors l'adresse de début de la zone BASIC (ce n'est pas gênant pour l'instant : le programme étant sur une seule ligne, l'interprêteur ne fait pas usage de ce pointeur).
L'octet précédant la zone est ensuite forcé à zéro, ainsi que les deux premiers octets (premier lien de chaîne de la nouvelle zone). Ensuite ? Eh bien il vaut mieux terminer par un NEW avant que l'interprêteur commence à se poser la moindre question ...

Alain Pinaud

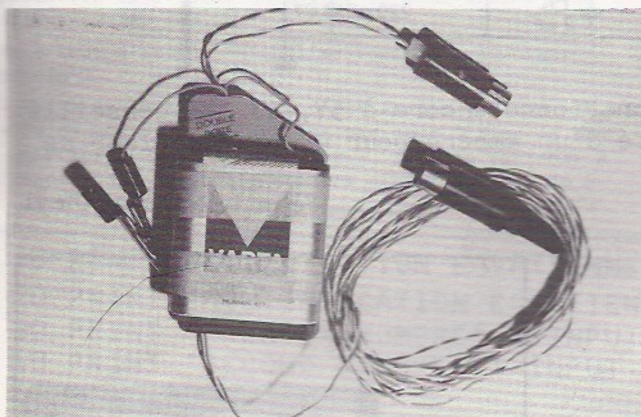
6

Modèle 1, modèle 3, 20 francs pour les connecter



Nous avons trouvé par miracle, au fond d'un tiroir du T.R.C. (TRACE Research Center) un prototype historique. Mieux vaut vous expliquer tout de suite son histoire AVANT que vous ne voyiez la photo de trop près ... !

Cela se passait il y a quelques mois. C'était un dimanche. Un TRS 80 modèle 3 venu tout droit des Etats-Unis (il n'y en avait pas encore en France), vint nous rendre visite. Impatients d'essayer les programmes dont nous disposions sur notre vieux modèle 1, nous nous heurtâmes au problème des cassettes dont souffrait encore notre système. C'était pourtant le seul lien possible entre les deux machines. Nous décidâmes alors de réaliser une liaison "sans cassettes" avec ce que nous avions, c'est-à-dire pas grand'chose : une pile de 4,5 volts, deux prises DIN, cinq résistances, un condensateur, deux transistors, et ... une boîte de cachous ! Le montage fonctionne encore et il est resté tel quel depuis ce jour.



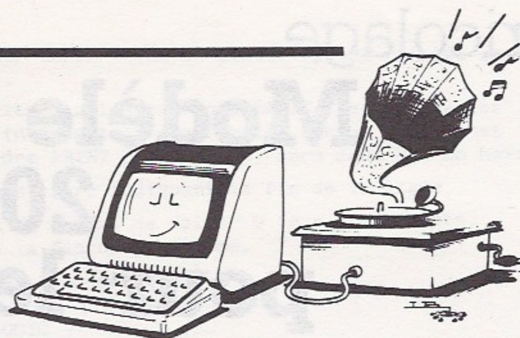


Ce prototype est "half-duplex", c'est-à-dire qu'il ne permet de transmettre des informations que dans un sens à la fois. Pour changer de sens, il faut permuter les deux prises ou employer un double inverseur. Avec cette interface, on peut aussi relier deux ordinateurs de même type (2 modèles 1 ou 2 modèles 3).

L'utilisation est des plus simples : en BASIC, il suffit de frapper les commandes CLOAD côté récepteur et CSAVE côté émetteur. Pour les programmes en langage machine, il faut faire SYSTEM côté récepteur et employer côté émetteur un utilitaire capable d'écrire des cassettes binaires (DUPLIK ou LMOFFSET par exemple).

Alain Pinaud

Sonorisez votre ordinateur

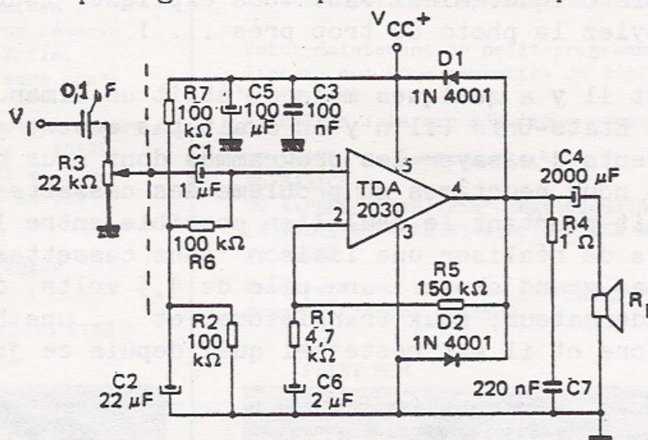


De plus en plus, les programmes de jeux disponibles sue les TRS-80 modèles 1 et 3, sont sonores, c'est-à-dire qu'ils sont capables d'émettre des sons divers et variés sur la sortie réservée habituellement au cassettophone.

Afin de ne pas "bricoler" son lecteur de cassettes ou de monopoliser la chaîne stéréo familiale, il est préférable de réaliser (ou d'acheter) un petit amplificateur qui sera réservé à cet usage. On trouve en effet des circuits intégrés amplificateurs basse fréquence très simples à mettre en oeuvre et fonctionnant à coup sûr.

Personnellement, j'ai choisi le TDA 2030 de Thomson-Efcis, dont la fiche technique nous donne une implantation sur circuit imprimé.
Il ne s'agit bien évidemment pas d'un amplificateur haute fidélité, mais la qualité de reproduction ne me paraît pas primordiale dans cette application.
De plus, ce circuit intégré est à l'épreuve des courts-circuits et possède une détection thermique qui permet de le protéger si la puissance dissipée est trop importante.

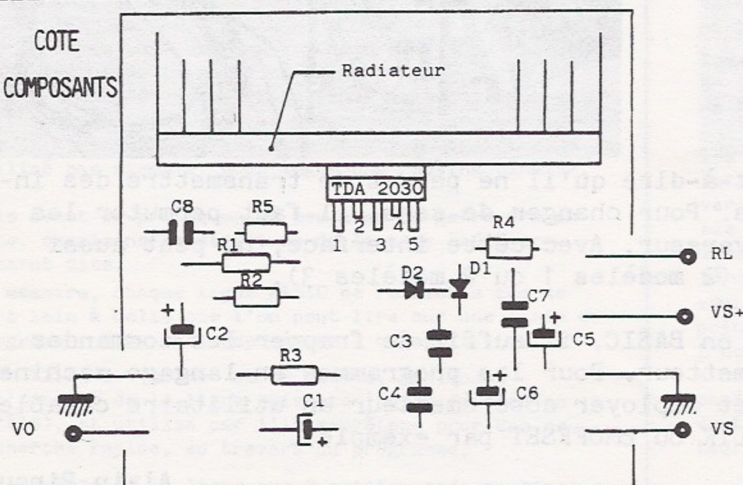
Schéma de principe :

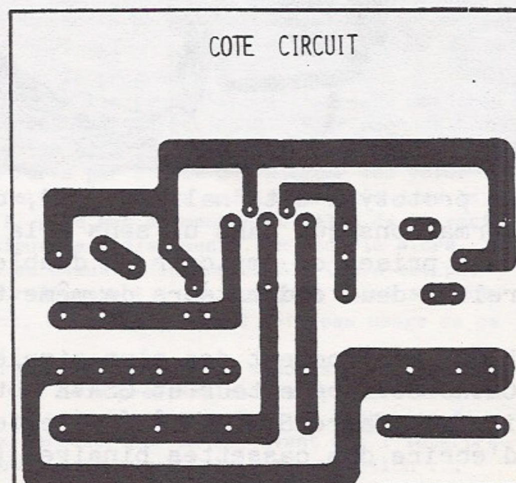


Pour l'alimentation, et bien que le circuit fonctionne encore à 5V, il est conseillé d'utiliser du 12V pouvant fournir une intensité d'un ampère pour un haut-parleur de 4 ohm (0,5 A pour un haut-parleur de 8 ohm).

Jean Lech Saconney.

Implantation du circuit imprimé :





La multiplication égyptienne



Cette méthode venue de l'antiquité, permettait aux anciens d'effectuer rapidement la multiplication de deux nombres. Au départ, elle nécessite de savoir :

- reconnaître un nombre pair d'un nombre impair,
- faire une addition,
- faire une division entière par 2.

Bien entendu, l'ordinateur sait effectuer des multiplications, ou tout au moins, il en donne l'impression. En réalité, l'algorithme de calcul qu'il utilise est peut être dérivé de cette méthode millénaire ... car en fin de compte, le microprocesseur lui, ne possède pas d'instruction de multiplication : il ne sait qu'additionner.

Nous allons donc simuler le mécanisme de la multiplication Egyptienne en BASIC, en nous aidant d'un organigramme.

Mais tout d'abord, de quelles fonctions avons-nous besoin ?

- Faire une addition ? Ça c'est facile : C = A + B.
- Faire une division entière par deux ? Rien de bien difficile : A = INT (B / 2). Ici, la variable A contiendra la partie entière (fonction INT) de B divisé par 2.
- Reconnaître si un nombre est pair ou impair ? Il y a plusieurs façons de procéder. Mais tout d'abord qu'est-ce qu'un nombre pair ? Nous pouvons dire que c'est un nombre qui, divisé par deux, ne donne pas de reste.

Par exemple : 3 divisé par 2 donne 1 comme diviseur ENTIER et 1 comme RESTE. 3 n'est pas pair. 6 divisé par 2 donne 3 comme diviseur ENTIER et 0 comme reste. 6 est pair.

En d'autres termes, si :

N - 2 * INT (N / 2) donne 0, le nombre est pair.

En BASIC, cela donne :

IF N - 2 * INT (N / 2) = 0 THEN ... nombre pair

Il y a une autre méthode qui fait appel au calcul binaire : si un nombre positif est pair, son dernier bit est 0, alors que s'il est impair, ce dernier bit est 1.

Par exemple, le nombre six (pair), s'écrit 0110 en binaire (dernier bit = 0). Le nombre 7 (impair) s'écrit 111 (dernier bit = 1).

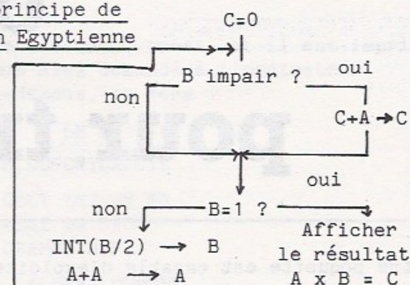
En BASIC, l'opérateur logique AND (appelé aussi ET logique ou intersection) permet d'effectuer un "masque" sur un nombre binaire. Il est alors facile d'isoler le dernier bit d'un nombre (masque = 1), pour en connaître la valeur :

```
nombre binaire  --→  10110101
masque          --→         1
opération AND   --→  --------
résultat        --→         1
```

En BASIC, cela s'écrit :
IF (N AND 1) = 0 THEN ... nombre pair
Vous êtes peut être en train de penser "mais ici, le nombre N n'est pas représenté en binaire ?". En fait, cela n'est pas nécessaire car l'opérateur AND, qui n'opère que sur des valeurs binaires, effectue implicitement cette conversion.
La seule restriction de cette dernière méthode, est que l'opérateur AND ne travaille que sur des nombres entiers au sens du BASIC, c'est-à-dire inférieurs à 32768.

Jean-Pierre du Tillet

Organigramme de principe de la multiplication Egyptienne




```
10 REM "MULTIPLICATION EGYPTIENNE POUR POQUETTE"
20 PRINT "MULTIPLICATION EGYPTIENNE"
30 PRINT "PREMIER NOMBRE" : INPUT C
40 PRINT "SECOND NOMBRE" : INPUT D
50 REM "CALCUL"
60 A=C : B=D
70 E=0
80 IF B-2*INT(B/2)=1 THEN E=E+A
90 IF B=1 THEN 140
100 B=INT(B/2)
110 A=A+A
120 GOTO 80
130 REM "EDITION DU RESULTAT"
140 PRINT "RESULTAT:" ; C ; "X" ; D ; "=" ; E
150 REM "CALCUL SUIVANT"
160 GOTO 20
```

POQUETTE

```
10 ' Programme de multiplication selon          ' F6
11 ' la methode dite "Egyptienne"               ' F5
12 '                                            ' F4
13 CLS                                          ' 6F
14 '                                            ' F2
15 '---------- entrees des donnees              ' F1
16 '                                            ' F0
17 PRINT TAB(15) "MULTIPLICATION EGYPTIENNE"    ' 88
18 PRINT TAB(15) STRING$(25,CHR$(131)) : PRINT  ' 8D
19 INPUT "PREMIER NOMBRE" ; C1                  ' 9A
20 INPUT " SECOND NOMBRE" ; C2                  ' F0
21 '                                            ' EB
22 '---------- algorithme de calcul             ' EA
23 '                                            ' E9
24 A=C1 : B=C2                                  ' 98
25 C=0                                          ' 9F
26 IF (B AND 1)=1 THEN C=C+A                    ' 88
27 IF B=1 THEN 34                               ' DD
28 B=INT(B/2)                                   ' 60
29 A=A+A                                        ' 7E
30 GOTO 26                                      ' ED
31 '                                            ' E1
32 '---------- edition du resultat              ' E0
33 '                                            ' DF
34 PRINT "RESULTAT:" ;C1; "X" ;C2; "=" ;C      ' CA
35 '                                            ' DD
36 '---------- calcul suivant                   ' DC
37 '                                            ' DB
38 GOTO 17                                      ' E5
```



utilitaire

Une méthode pour tracer vos courbes

Votre poquette est capable d'exploiter des chaines de caractères et il possède une imprimante. C'est bien. Mais peut-on lui faire tracer des courbes ? Oui. Ce petit programme vous le démontrera.

Sur le poquette, les chaines de caractères sont difficilement utilisables comme telles, mais il est toutefois possible de détourner le problème en considérant une chaîne comme la somme de plusieurs sous-chaines.

Une variable contenant ici 16 caractères (nombre de caractères par ligne imprimée) peut être divisée en 16 variables alphanumériques, pour lancer sur l'imprimante le dessin d'une courbe.

Voici le programme :

```
10 FOR W=1 TO 16
20 A$(W)=" "
30 NEXT W
40 FOR V=1 TO 40
50 S=INT (SIN(V*10)*(2 ))+10
60 A$(S)="+"
70 PRINT A$;B$;C$;D$;E$;F$;G$;H$;
         I$;J$;K$;L$;M$;N$;O$;P$
80 A$(S)=" "
90 NEXT V
```

Essayez-le et vous aurez une jolie corbe sinusoïdale ! Ne pas oublier de mettre la machine en mode DEGRE.

En modifiant la ligne 50, vous obtiendrez la représentation d'une courbe de votre choix. Dans ce cas, vous devrez également modifier la ligne 40 selon l'incrément de l'ordonnée que vous désirez.
En découpant ainsi une chaîne de caractères en sous-chaînes, il est possible de la décomposer pour obtenir (pourquoi pas ?) les mêmes résultats qu'avec les puissantes instructions LEFT$, RIGHT$ et MID$ du modèle 1.

Michel Claveau

utilitaire

Un mini traitement de textes

Vous connaissez tous le programme Scripsit sur modèles 1, 2 ou 3 ? En voici une version (bien moins puissante et bien moins "intelligente") pour le poquette TRS-80 / Sharp PC-1211.

La tâche qui vous incombe consiste à entrer la date, le nom du destinataire, le texte et la signature, le tout dans un ordre quelconque puisque effectué en mode DEF.
Il est possible d'éteindre l'ordinateur à n'importe quel instant, les dernières données seront sauvegardées : vous pouvez par exemple, entrer la date et le nom du destinataire, puis éteindre la machine. Plus tard, vous pourrez l'allumer de nouveau puis entrer la lettre et la signature.
Il est aussi possible de sauvegarder des "lettres" sur cassettes.
Enfin, vous pouvez imprimer la lettre à n'importe quel moment, même si vous avez éteint l'ordinateur auparavant.

```
 5:CLEAR :BEEP
   1:INPUT "SCR
   IPSIT/POCKET
   ",A
 6:GOTO 5
10:"N"INPUT "NO
   M :";A$,B$
15:END
20:"D"INPUT "DA
   TE :";C$:END
25:"L":D=7
30:D=D+2:INPUT
   "CARACS.:";A
   $(D):BEEP 1:
   E=D+1:INPUT
   "CARACS.2:";
   A$(E):GOTO 3
   0
35:END
40:"="PRINT "
      LE ";C$:
   PRINT " ":
   PRINT "M.";A
   $;B$:PRINT "
    "
50:FOR E=9TO D
   STEP 2:F=E+1
   :PRINT A$(E)
   ;A$(F):NEXT
   E:PRINT " ":
   PRINT "  ";G
   $;H$:PRINT "
    ":PRINT " "
   :PRINT " ":
   END
60:"S"INPUT "SI
   GNATURE :";G
   $,H$
```

Mode d'emploi

Passez en mode DEF et effectuer :

SHFT D = date
- Affichage : DATE :
- Entrez une date selon la forme : JJMM/AA
- Exemple : 0112/82
- Affichage : DATE : 0112/82
- Appuyez sur ENTER
- La date est entrée en mémoire.

SHFT N = nom
- Affichage : NOM :
- Entrez un nom
- Exemple : DUPONT
- Appuyez sur ENTER ENTER
- Pour les noms plus longs, faites :
- DUPONTL ENTER
  1234567
- Affichage : ?
- Entrez : AJOIE ENTER
- Vous pouvez ainsi entrer jusqu'à 14 caractères par nom, de cette manière.

SHFT L = la lettre elle-même

Affichage : CARACS. :
Entrez les 7 premiers caractères de votre lettre
Soit à imprimer, par exemple, le message suivant :

CECI EST UN ESSAI DU PROGRAMME SCP.

Entrez : CECI ES ENTER
1234567

Bip ! Ce son signifie que l'on passe sur la seconde moitié du rouleau (à cause de la taille des mémoires, j'ai dû partager la ligne en deux : 7+7).

Affichage : CARACS.2:
Entrez ensuite : T UN ES ENTER
1234567

Le mot ESSAI est coupé ; la suite sera sur la ligne suivante.
Affichage : CARACS.:
Entrez : -SAI DU ENTER
1234567

(bip !)
Et ainsi de suite ...
CARACS.2: PRO- ENTER
12345
CARACS.: GRAMME ENTER
1234567

(bip !)
CARACS.2: SCP. ENTER
1234

Faites ENTER à la fin de la lettre.

SHFT S = signature

Affichage : SIGNATURE :
Entrez : ALBERT ENTER
1234567

Affichage : ?
Entrez : DUMONT ENTER
123456

Pour les noms plus courts, faites 2 fois ENTER

Pour imprimer, (enfin arrive le résultat tant attendu !), ne pas oublier de mettre l'imprimante en route ...

SHFT =

Tout se passe automatiquement, et il est imprimé tout ce que vous avez demandé à l'ordinateur.
Dans l'exemple ci-dessus, ce sera :

```
          LE 0112/82

     M.DUPONTLAJOIE

     CECI EST UN ES
     -SAI DU PRO-
     GRAMME SCP.

          ALBERT DUMONT
```

Le programme cadre automatiquement la date, le nom, la lettre et la signature.
L'utilisation semble un peu compliquée mais, avec un peu d'habitude, cela devient très facile - et très pratique.

Quelques petits conseils maintenant :

. Pour imprimer plusieurs lettres à la suite, il suffit d'appuyer de nouveau sur SHFT = autant de fois que vous le désirez.
. Ne pas oublier de faire CLEAR avant de commencer une nouvelle lettre.
. Pour sauvegarder la lettre sur cassette, il suffit de faire : PRINT ≠ "LETTRE".
. Pour relire la lettre à partir de la cassette, faire : INPUT ≠ "LETTRE" et SHFT = pour l'imprimer (ne pas oublier de mettre la date à jour si nécessaire !).

Ecrivez de bons "romans" !!

Jean David Blanc

Des cavernes dans le poquette



- Vous vous trouvez dans un tunnel mal éclairé, le tunnel continue vers le nord et se termine au sud, vous voyez plusieurs portes sur le côté, il y a un couteau et une vieille corde près d'une des portes. Vous voyez un lion ...
- Je prends la corde.
- O.K.
- Au nord
- vous vous trouvez dans une large pièce ...

Qui ne connaît pas les programmes d'aventures ? Ce sont des sortes de labyrinthes que l'on explore à la recherche de trésors et d'action.
Il en existe un grand nombre pour le TRS-80, mais leur taille et leur complexité les interdisent aux petits systèmes. En effet, de tels programmes demandent un minimum de 4K octets de mémoire... Un peu trop pour le poquette !

Le programme décrit ci-dessous ne fonctionne pas comme les programmes d'aventures habituels : au lieu de contenir toutes les pièces au départ, il les "invente" au fur et à mesure, ce qui permet d'explorer aussi longtemps que l'on veut, des cavernes obscures et dangereuses.

Mode d'emploi

La première rangée des touches programmables du poquette ont été assignées ainsi :

- A : pour démarrer le jeu. Lorsque le texte "NO.=" apparaît, répondez un nombre quelconque pour initialiser le programme.
A la question "NIVEAU(0/5)", répondez par un nombre correspondant au niveau de difficulté souhaité dans cette aventure. Le programme vous décrira alors la première pièce, puis - éventuellement - le monstre qui l'occupe, ainsi que les différentes sorties dans les quatre directions (est, ouest, nord et sud).
Lorsque le texte "ORDRE" sera affiché, vous pourrez frapper votre commande (SHFT puis l'une des dix touches programmables).

- S : Déplacement vers le nord. S'il y a un monstre dans le pièce, il se peut que vous ne puissiez passer. Dans ce cas, il vous faudra essayer une autre issue. Si vous ne pouvez plus sortir, alors vous avez perdu.

- D : Même chose pour aller à l'est.

- F : Pour le sud.

- G : Pour l'ouest.

- H : Pour vous débarrasser des monstres qui vous gènent dans votre progression, il existe dix anneaux "magiques" ayant le pouvoir de tuer certains monstres. Ces anneaux sont dispersés dans les cavernes sauf l'anneau numéro 1 que vous possédez au départ. Le but du jeu est, en fait, de retrouver les autres anneaux. Si vous en apercevez un, SHFT H vous permettra de le saisir.

- J : Donne l'inventaire des anneaux en votre possession.

- K : Pour tuer un monstre. Ceci donne un point à votre score et vous permet de changer de pièce si le monstre vous en empêchait. Pour cela, il vous faut possèder l'anneau correspondant à ce monstre (sinon, cela ne marche pas !).

- L : Donne le score lorsque vous désirez abandonner la partie (parfois, vous ne pouvez pas faire autrement !).

- = : Pour revoir la description d'une pièce.

Pour entrer le programme, il faut d'abord charger les DATA dans les mémoires correspondantes, à l'aide par exemple d'une petite ligne de BASIC que l'on effacera ensuite :

```
10 FOR G=22 TO 65 :
INPUT A$(G) :
NEXT G : END
```

L'ensemble DATA et programme occupent toute la mémoire. N'essayez donc pas d'en ajouter davantage !

Description des mémoires

A à U : mémoires de manoeuvre.
A(12) à A(21) : endroits où se trouvent les dix anneaux. A(12) sera toujours à 0, puisqu'il est porté par le joueur.
A(22) à A(26) : noms des différentes sorties (PORTE, CHEMIN, etc ...).
A(27) à A(46) : suite d'adjectifs divers (VIEUX, SALE, etc ...).
A(47) à A(56) : noms des pièces (VOLCAN, TEMPLE ...).
A(57) à A(61) : différentes "bestioles" qui peuplent les cavernes du poquette.
Enfin, A(62) à A(65) contiennent les quatre directions et A(66) à A(69) sont utilisées par le programme.

Il est évidemment possible de changer le contenu des mémoires 22 à 61 (que diriez-vous d'une autre aventure à bord d'une fusée, par exemple, en remplaçant "TUNNEL" par "CABINE" et "DRAGON" par "ALIEN"...?).

Si vous possédez une imprimante, vous aurez même une trace écrite de vos aventures ...

Si vous enregistrez le programme, n'oubliez pas de sauver les DATA avec : PRINT # "(le titre) ";A

Le niveau de jeu règle la facilité de passage d'une pièce à l'autre (ou le nombre de chemins qui vous seront accessibles).

(listing au dos)

Charles Feydy

A	S	D	F	G	H	J	K	L	=
DEBUT	NORD	EST	SUD	OUEST	PREND	INVENT	TUER	FIN	REGAR
Z	X	C	V	B	N	M	SPC		ENTER

PROGRAMMES ET MEMOIRES.

```
10:"A"INPUT "NO.= ";D,"NIVEAU(0/5)= ";F:F=4F
20:FOR G=15TO 21:E=D+15G:GOSUB "R":A(G)=INT E:NEXT G
30:C=5:A(67)=0:GOSUB 180:GOTO E2
35:A$(66)="":IF C>E2LET C=C-E2*INT (C/E2
40:G=B-INT B:FOR H=1TO 3:G=G*10:A(8+H)=INT G+7+(H+1)*10:G=G-INT G:NEXT H
50:BEEP 1:PRINT "VOUS ETES DANS":PRINT "UN ";A$(I);" ";A$(K);" ";A$(J)
60:IF INT A>40-FLET G=57+INT B/20:PRINT "IL Y A UN ";A$(G):A$(66)=A$(G)
70:FOR G=1TO 10:IF A(11+G)=CPRINT "VOUS TROUVEZ":PRINT "L-ANNEAU #";G
75:NEXT G:PRINT "IL Y A:
80:FOR G=1TO 7 STEP 2:A$(69)="":IF G=7 LET A$(69)=" "
85:K=G:GOSUB "T":IF K>=4LET K=G+1:J=K/2+61:GOSUB "T":K=22+INT (K/2:PRINT "UN ";A$(K);" A";A$(69);A$(J)
90:NEXT G:RETURN
100:INPUT "ORDRE ",G
110:GOTO E2
120:"S"K=1:GOTO 160
130:"D"K=3:GOTO 160
140:"F"K=5:GOTO 160
150:"G"K=7
160:G=K:A(68)=K:GOSUB "T":IF K<4GOTO 270
170:K=G+1:GOSUB "T":IF K/2-INT (K/2)<>0 IF INT A>40-FPRINT "LE MONSTRE VOUS ARRETE":GOTO E2
180:FOR G=1TO 2:C=C-4+A(68):E=C+D:GOSUB "R":A(G)=E:NEXT G:GOSUB 35:GOTO E2
190:"H"FOR G=12 TO 21:IF A(G)=CPRINT "OK.":A(G)=0
200:NEXT G:GOTO E2
210:"J"I=0:PRINT "VOUS AVEZ":FOR G=12TO 21:IF A(G)=D LET H=G-11:PRINT "L-ANNEAU #";H:I=1
220:NEXT G:IF I=0PRINT "RIEN
225:GOTO E2
230:"K"IF A$(66)=""GOTO E2
240:G=INT B-10*INT (B/10:IF A(G+12)=0 PRINT "OK.":A(67)=1+A(67:A$(66)="":FOR H=2TO 8 STEP 2:K=H:GOSUB "T
250:IF A(G+12)=0 IF K/2-INT (K/2)<>0LET A=A+1/10^H
260:IF A(G+12)=0 NEXT H:GOTO E2
270:PRINT "NON!":GOTO E2
280:"T"K=(INT (A*10^K)-(10*INT (A*10^(K-1:RETURN
290:"R"E=ABS (SIN E*E2:RETURN
300:"L"PRINT "SCORE: ";A(67):END
400:"="GOSUB 35:GOTO E2
```

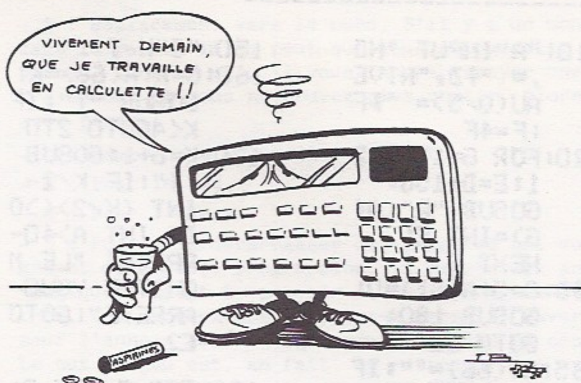
Touches programmables

NOTE : faire d'abord CLEAR, puis introduire les DATA de A$(22) à A$(65) par une ligne BASIC que l'on effacera après usage (voir plus haut). Entrer ensuite le programme. Remarquez que ce dernier a été concentré au maximum (GOTO E2 au lieu de GOTO 100, par exemple).

```
22.  E PORTE
23.   CHEMIN
24.  E ROUTE
25.   TUNNEL
26.  E TRAPPE
27.   VIEUX
28.   SALE
29.   GRAND
30.   PETIT
31.   TRISTE
32.   SOMBRE
33.   ETRANGE
34.   LARGE
35.   ANCIENT
36.   LARGE
37.   ROMAIN
38.   GREC
39.   PERDU
40.   DESERT
41.   HUMIDE
42.   SOMBRE
43.   EN BOUE
44.   BRULE
45.   MAUDIT
46.   EN LAVE
47.   TUNNEL
48.   TEMPLE
49.   TOMBEAU
50.   CAVEAU
51.   LOCAL
52.   COULOIR
53.   TUNNEL
54.   PUITS
55.   TUNNEL
56.   VOLCAN
57.   DRAGON
58.   SORCIER
59.   TROLL
60.   LION
61.   VAMPIRE
62.  U NORD
63.   L-EST
64.  U SUD
65.   L-OUEST
```

POQUETTE

Quand votre poquette re-devient calculette



Qui d'entre nous n'a jamais eu à additionner de longues colonnes de nombres ? Souvenez-vous ! En vérifiant, vous ne trouviez pas la même valeur. A la deuxième vérification un résultat encore différent apparaissait sur votre calculatrice. Dans l'effarement qui suivait vous doutiez de tout : des piles de votre machine, de votre capacité à lire, de l'hygrométrie de la pièce.
Il ne vous restait plus alors, pour éclaircir le problème, qu'à vérifier votre biorythme, vos dons télékinésiques (qui auraient pu influer sur la mémoire de votre calculatrice), ou faire brûler un cierge, en espérant que cela s'arrange.
Aujourd'hui la solution arrive ! Soudain, dans l'éclat merveilleux de "TRACE" une apparition fantastique se montre : un programme d'addition sur la très jolie TRS-80 POCKET. Fini les crises de dépression ! Ne fouettez plus votre calculatrice ! Dormez enfin en paix !

LE PROGRAMME

```
10 "D":CLEAR:E=1
20 "S":D=1:A=E:GOTO 60
30 "C":D=2:GOTO 50
40 "V":D=3
50 INPUT"NUM.DEPART ";A:IF A<1 GOTO 50
55 IF A>=E GOTO 50
60 IF D>1 PRINT A,A(A+5):IF D=3 LET A=A+1:GOTO 55
70 INPUT B:C=C-A(A+5)+B:A(A+5)=B:A=A+1:IF A>E LET E=A
80 GOTO 60
90 "F":A=A(E+4):B=E-1:PRINT"NB:";B;" TOTAL:";C:
   PRINT"DERNIER :";A:GOTO 90
```

Dans le programme les variables suivantes sont utilisées :

A = Compteur
B = Mémoire-tampon pour la saisie
C = Total
D = Code opération (1.saisie 2.correction 3.visualisation)
E = Nombre de données rentrées
Les données sont stockées dans A(5), A(6), A(7),...

Le programme occupe 221 pas. S'il est le seul en mémoire, il vous permettra d'additionner 170 valeurs (149 mémoires libres + 26 mémoires fixes - 5 variables de contrôle).

UTILISATION

Le programme doit être utilisé <u>en mode DEF</u>, afin de pouvoir se servir des étiquettes incluses dans le programme.

INITIALISATION. Il suffit de faire 'SHFT D'. Cela efface les données présentes et initialise le compteur et le code opération. On fera donc l'initialisation à chaque nouvelle addition.

Le programme demande ensuite les valeurs, que l'on pourra saisir immédiatement. Si l'on tape 'Enter', sans avoir auparavant tapé de chiffres, la saisie sera refusée, et la question reposée.

RESULTATS. En faisant 'Shft F' on visualisera le nombre de données et le total. Si on tape ensuite 'Enter' la dernière valeur s'affichera.

SAISIE. 'Shft S' permettra de retourner en saisie. Les nouvelles valeurs seront rajoutées à la suite des données déjà rentrées.

VISUALISATION/ 'Shft V' vous servira à visualiser les données rentrées, et ce à partir d'un numéro de départ saisi (il doit être compris entre 1 et le nombre de données rentrées, sans quoi il sera refusé). 'Enter' après une valeur permet de visualiser la suivante, jusqu'à la dernière, le numéro de départ étant ensuite redemandé.

CORRECTION. 'Shft C'. Le fonctionnement est identique à la visualisation, à la seule différence qu'après avoir visualisé une valeur, le fait de taper sur 'Enter' vous permettra de saisir la nouvelle donnée. Le remplacement et le nouveau total sont calculés immédiatement. La saisie de 'Enter' sans avoir tapé de chiffres pour le remplacement de la valeur est refusée.

DIVERS

Chaque fois qu'un point d'interrogation apparaît, une valeur numérique doit être rentrée (dans tous les cas 'Enter' seul est refusé).
On passe d'une opération à une autre par 'Shft' et une lettre.
Il est possible d'arrêter le TRS-80 POCKET, puis de le remettre en route sans perdre ni les données, ni les variables de contrôle.
Les étiquettes ont été choisies pour leur groupement, et leur mnémonique :

D . Début (ou départ)
S . Saisie
C . Correction
V . Visualisation
F . Résultat (en dessous du R, car celui-ci n'est pas une touche réservable).

```
 off on

 q  w  e  r  t  y
 a [S][D][F] g  h
 z  x [C][V] b  n
```

Michel Claveau

BASIC 2 BLINK BOOT BREAK BUILD

Le Modèle 1 possède plusieurs Systèmes d'Exploitation des Disquettes (SED, ou DOS en anglais). Ces différents systèmes ont beaucoup de points communs, particulièrement ceux de la "lignée" TRSDOS-NEWDOS-NEWDOS80-LDOS, qui s'inspirent tous, à quelques détails près, du TRSDOS de Tandy (version initiale 2.1). Nous publierons régulièrement sous forme de fiches pratiques, les commandes de ces différents DOS, classées alphabétiquement. Ces fiches ne prétendent pas remplacer les manuels originaux qu'il est impératif de posséder (il n'était pas envisageable de publier sous cette forme l'équivalent de plusieurs centaines de pages ...).

Leur but est de fournir à l'utilisateur une information rapide d'accès (classement alphabétique), pratique (exemples d'utilisation), comparative (différents DOS) et dans une langue qu'il est censé connaître (français)...

Les DOS les plus répandus ont été retenus pour constituer ces fiches. Ce sont le TRSDOS version 2.3 de Tandy, le NEWDOS80 version 2 d'Apparat et le LDOS version 5.0 de Lobo Drives International. En règle générale, nous nous baserons sur les commandes du TRSDOS, en signalant les différences ou les apports rencontrés dans les autres systèmes d'exploitation.

Conventions d'écriture

Dans la description des commandes, les mots en caractères majuscules désignent les commandes proprement dites, ainsi que les mots-clés associés. Les mots en caractères minuscules ne font pas partie des commandes mais en représentent la forme. Au moment de l'exécution, il faudra leur substituer le ou les mots appropriés. Parmi ceux-ci, nous trouverons par exemple : "nomfich", qui désigne un nom de fichier, "motpasse" qui signifie qu'un mot de passe (password) est attendu à cet endroit. Dans tous les cas, un coup d'oeil sur les exemples donnés devrait lever toute ambiguité.
Enfin, il faut signaler que le NEWDOS80 emploie indifféremment la virgule ou l'espace comme séparateur dans les commandes.

BASIC2

But : commande employée par TRSDOS ou NEWDOS80 pour exploiter le BASIC niveau 2 du TRS80. Après exécution, le DOS est hors d'usage (correspond à une mise sous tension avec la touche BREAK enfoncée).

BLINK paramètre

Paramètre : absent ou Y
N

But : commande employée par NEWDOS80 pour caractériser le curseur. Si le paramètre est Y (ou est absent) le curseur clignote (blink en anglais). Si le paramètre est N, le curseur est fixe

```
Exemple : BLINK,Y      (le curseur clignote)
          BLINK,N      (le curseur est fixe)
```

BOOT

But : pour LDOS et NEWDOS80, correspond à un appui sur le bouton RESET. Le premier secteur de la disquette est chargé en mémoire (bootstrap) et est exécuté. Il s'en suit une ré-initialisation complète du système.

Durant la phase de ré-initialisation (mise sous tension, bouton RESET ou ordre BOOT), deux touches du clavier sont consultées : ENTER et 'flèche haute'.
Si ENTER est enfoncée, l'ordre AUTO est ignoré (voir AUTO).
Si 'flèche haute' est enfoncée, le "driver" clavier n'est pas intercepté par le DOS (seule la routine ROM est utilisée). L'anti-rebonds du clavier ainsi que la fonction répétition automatique des touches ne sont donc pas pris en compte. Cela peut s'avérer nécessaire pour l'exécution de certains programmes venant recouvrir une partie du DOS (écrasement probable de la routine anti-rebonds).

De plus, avec LDOS, un appui sur la touche CLEAR empèche l'exécution du configurateur automatique (voir commande SYSTEM). Cette touche doit aussi être maintenue avec 'flèche haute' pour éviter l'interception du driver clavier.
Avec LDOS, on trouve d'autre part une possibilité d'appel automatique du DEBUG (outil de mise au point) si la touche D est maintenue enfoncée.

BREAK paramètre

Paramètre : absent ou Y
N

But : Cette commande NEWDOS80 permet de valider (paramètre Y) ou d'invalider (paramètre N) l'effet de la touche BREAK. L'invalidation est annulée dès le retour au prochain message "DOS READY".

Exemple :

```
BREAK,N
BASIC
```

sera sans effet (DOS READY après la première commande). Il faudra faire :

```
BASIC
CMD "BREAK,N"
```

et cette fois, la touche BREAK sera bien invalidée dans BASIC.

BUILD nomfich (paramètre)
Paramètre : APPEND HEX

But : cette commande LDOS permet de construire facilement un fichier de commandes (voir CHAIN et AUTO) sur disque, à partir du clavier (ou tout autre organe assigné à ce dernier). Si l'extension du nom de fichier n'est pas spécifiée, ce sera implicitement '/JCL'. Si l'on ne désire pas donner d'extension, il suffira de faire suivre le nom de fichier du symbole '/'.
Le fichier est constitué à partir du clavier, ligne par ligne (255 caractères max par ligne), chaque ligne étant terminée par un appui sur la touche ENTER.
Soit à exécuter par exemple, le chainage automatique suivant :

```
DIR
BASIC
LOAD "PROG"
```

le nom "FICH" étant donné à ce fichier de commande sur disque, il suffira de frapper :

```
BUILD FICH      (enter)
DIR             (enter)
BASIC           (enter)
LOAD "PROG"     (enter)
(break)
```

Notez que la touche BREAK clôt le fichier et déclenche sa sauvegarde sur disque sour le nom "FICH/JCL".
Ce fichier pourra ensuite être exécuté par :

```
CHAIN FICH
```

Le fichier 'nomfich' ne doit pas être présent sur disque au moment de l'exécution de la commande BUILD, sauf si le paramètre APPEND est spécifié. Dans ce dernier cas, les lignes de commandes entrées après BUILD seront ajoutées au fichier 'nomfich' après appui sur la touche BREAK.
Si le paramètre HEX est utilisé, les caractères composant chaque ligne sont entrés sous forme hexadécimale (00 à FF), et non plus sous forme ASCII. Cela permet ainsi d'inclure des codes binaires (graphiques par exemple) dans le fichier de commande. Chaque octet doit être entré sous forme de deux digits hexadécimaux (00 à FF) sans espaces séparateurs, et chaque ligne doit être terminée par 'OD' (code du retour chariot), la touche ENTER étant sans effet sur le contenu du fichier.

Si nous voulons nous compliquer la vie... et ajouter la commande RUN à notre précédent progrmame, nous pourrons faire :

```
BUILD FICH (APPEND,HEX)     (enter)
52554E0D                    (enter)
(break)
```

Index des mots-clés

La liste des mots-clés figurant ci-dessous est destinée à être exploitée par le programme de gestion des articles décrit dans le numéro 1 de TRACE (pages 58 à 64). Le format est le suivant:

Nom abrégé de l'article, numéro de page : 1 à 4 mots-clés.

En prenant l'habitude d'entrer cette liste à chaque numéro de TRACE, vous constituerez ainsi une petite base de données vous permettant de trouver plus efficacement et plus rapidement l'article que vous cherchez. Bien entendu, cette liste est donnée à titre indicatif, et vous pourrez toujours en modifier ou en compléter le contenu, afin de l'adapter à vos besoins ; mais avant toute chose, vous devez vous imposer une certaine normalisation des mots-clés (en principe, ils sont limités à 12 caractères) PROGRAMME par exemple, indiquera toujours qu'un listage de programme est donné dans l'article. Pour cela aussi, faites-nous part de vos idées!

LNW-80, 17: LNW80, COMPATIBLES, MATERIEL
MODELE 16, 19: MODELE16, 68000, MATERIEL
POQUETTES, 19: SHARP1500, PC2, MATERIEL
ESSAI EDTASM+, 20: EDTASM, ASSEMBLEUR, ESSAI, LOGICIEL
ESSAI ADVENTURE, 21: AVENTURE, JEU, ESSAI, LOGICIEL
LIBRAIRIE, 22: LIVRES, ESSAI
SOMME DE CONTROLE, 27: CONTROLE, ERREUR, LOGICIEL, PROGRAMME
DANS LE BATIMENT, 31: APPLICATION, PROF, VISICALC
DANS LES TRANSPORTS, 33: APPLICATION, PROF, MODELE1, OPTIQUE
PROGRAMMES D'ENSEIGNEMENT, 35: APPLICATION, ENSEIGNEMENT, PROGRAMME
OLYMPIC DECATHLON, 38: JEU, ESSAI, LOGICIEL
MUMATH, 39: ESSAI, LOGICIEL, LISP, MATHS
MATHIC, 42: JEU, PROGRAMME, ARITHMETIQUE
DIFFERENCES ENTRE 1 ET 3, 44: MODELE1, MODELE3, MATERIEL, LOGICIEL
PREVISIONS, 47: PROGRAMME, PREVISIONS, STATISTIQUES
NEVADA COBOL, 50: ESSAI, LOGICIEL, LANGAGE, COBOL
MAPPER CP/M, 52: ESSAI, MATERIEL, CP/M, MODELE1
VARIABLES EN MEMOIRE, 53: LOGICIEL, BASIC, VARIABLES
OUTILS, 55: OUTILS, BASIC, ASSEMBLEUR, MONITEUR
BOITE DE CACHOUS, 57: MATERIEL, COMMUNICATION, MODELE1, MODELE3
AMPLI BF, 56: BRICOLAGE, AMPLI, SON, MODELE1
MULTIPLICATION EGYPTIENNE, 59: POQUETTE, PROGRAMME, ARITHMETIQUE
TRACE DE COURBES, 60: POQUETTE, COURBES, CHAINES, PROGRAMME
MINI-TRAITEMENT DE TEXTES, 61: POQUETTE, TDT, PROGRAMME
AVENTURE, 62: POQUETTE, AVENTURE, PROGRAMME, JEU
ADDITIONNEUR, 64: POQUETTE, ADDITION, PROGRAMME, VERIFICATION
FICHE DOS A A Z, 65: FICHE, DOS

Imprimerie Laballery, 58500 Clamecy. Imprimé en France. Directeur de la publication Jean-Pierre Nizard
Dépôt légal : mai 1982 - Numéro de commission paritaire : en cours

Pour mieux choisir "votre" ordinateur et pour mieux l'utiliser.



Lisez

L'ORDINATEUR INDIVIDUEL

Vous y trouverez :

L'actualité et les tendances de l'informatique individuelle • des galops et des bancs d'essai des principaux matériels • des panoramas et des tests comparatifs • le point des grandes manifestations internationales • des articles d'initiation • des synthèses • des programmes • des interviews "exemplaires" • des conseils • des idées • des astuces.

L'ORDINATEUR INDIVIDUEL

chez votre marchand de journaux

41 rue de la Grange aux Belles - 75010 Paris